NASA/TM—2003–212236, Vol. 5, Ver. 1.0

GEOSAT Follow-On (GFO) Altimeter Document Series

Volume 5, Version 1.0

GFO Radar Altimeter Processing at Wallops Flight Facility

D.W. Lockwood
A.M. Conger

August 2003

The NASA STI Program Office ... in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) Program Office plays a key part in helping NASA maintain this important role.

The NASA STI Program Office is operated by Langley Research Center, the lead center for NASA's scientific and technical information. The NASA STI Program Office provides access to the NASA STI Database, the largest collection of aeronautical and space science STI in the world. The Program Office is also NASA's institutional mechanism for disseminating the results of its research and development activities. These results are published by NASA in the NASA STI Report Series, which includes the following report types:

- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA's counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.
- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.
- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.
- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or cosponsored by NASA.
- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and mission, often concerned with subjects having substantial public interest.
- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services that complement the STI Program Office's diverse offerings include creating custom thesauri, building customized databases, organizing and publishing research results . . . even providing videos.

For more information about the NASA STI Program Office, see the following:

- Access the NASA STI Program Home Page at http://www.sti.nasa.gov/STI-homepage.html
- E-mail your question via the Internet to help@sti.nasa.gov
- Fax your question to the NASA Access Help Desk at (301) 621-0134
- Telephone the NASA Access Help Desk at (301) 621-0390
- Write to:
  NASA Access Help Desk
  NASA Center for AeroSpace Information
  7121 Standard Drive
  Hanover, MD 21076–1320

NASA/TM—2003–212236, Vol. 5, Ver. 1.0

GEOSAT Follow-On (GFO) Altimeter Document Series

Volume 5, Version 1.0

GFO Radar Altimeter Processing at Wallops Flight Facility

Dennis W. Lockwood, A.M. Conger
Raytheon ITSS, Wallops Island, VA

National Aeronautics and
Space Administration

Goddard Space Flight Center
Greenbelt, Maryland 20771

August 2003

About the Series

The TOPEX Radar Altimeter Technical Memorandum Series is a collection of performance assessment documents produced by the NASA Goddard Space Flight Center Wallops Flight Facility over a period starting before the TOPEX launch in 1992 and continuing over greater than the 10 year TOPEX lifetime. Because of the mission's success over this long period and because the data are being used internationally to redefine many aspects of ocean knowledge, it is important to make a permanent record of the TOPEX radar altimeter performance assessments which were originally provided to the TOPEX project in a series of internal reports over the life of the mission. The original reports are being printed in this series without change in order to make the information more publicly available as the original investigators become less available to explain the altimeter operation and details of the various data anomalies that have been resolved.

Available from:

NASA Center for AeroSpace Information
7121 Standard Drive
Hanover, MD 21076–1320
Price Code: A17

National Technical Information Service
5285 Port Royal Road
Springfield, VA 22161
Price Code: A10

Foreword

This document is a compendium of the WFF GFO Software Development Team's processing of GFO CAL/VAL Data. It includes many elements of a Requirements Document, a Software Specification Document, a Software Design Document, and a User's Guide. In the more technical sections, this document assumes the reader is familiar with GFO and its CAL/VAL Data.

Table of Contents

List of Figures

List of Tables

Section 1

Introduction

1.1 Purpose

This document provides a detailed description of GEOSAT Follow-On (GFO) Radar Altimeter data processing at NASA Goddard Space Flight Center's Wallops Flight Facility (WFF).

1.2 Definition of Data

The GFO altimeter sensor data is downloaded from spacecraft memory several times per day to ground system remote sites, where raw telemetry data files containing the payload science and engineering data are constructed. The data files are then transferred to the Navy's Payload Operations Center (POC), one data file per DSU dump. Integral to the data files nomenclature is the file identifier of data and time, YYDOY_HH_MM_SS, signifying the beginning time of that data. These data files, segmented by date and time, are classified as segments. There can be any number from two to six segments of data per day.

The Navy's Altimetry Data Fusion Center (ADFC) electronically transfers to Wallops the data segments required for processing at WFF. The data consists of three types of data files: 1) set of four Cal/Val (ra_data,ra_cal_data,eng_data,wvr_data) files; 2) Sensor Data Record (SDR) file; 3) Navy Geophysical Data Record (NGDR) file.

Examples of Cal/Val data segments are:

eng_data03122_03_15_45.dat
ra_data03122-03_15_45.dat
ra_cal_data03122_03_15_45.dat
wvr_data03122_03_15_45.dat

1.3 Data Flow

GFO altimeter data are regularly pushed to WFF from the NAVY Altimeter Data Fusion Center (ADFC). The automated File Transfer System (FTS) at WFF, documented in Appendix F, handles the distribution of the data based on a number of configuration scripts, examples of which are contained in Appendix G. Once ingested, a number of processes are executed to produce summary reports, data trends, and stores averaged data in the local database management system (DBMS) for trend analysis.

Section 2

Related Documentation

2.1 Publications

- R.L. Brooks and D.W. Lockwood, *Temperature Correction for GFO Wind Speed,* May 2003. [http://gfo.wff.nasa.gov/docs.html]
- D. W. Hancock III, G.S. Hayne,R.L. Brooks, D.W. Lockwood, *GFO Altimeter Engineering Assessment Report - From Launch to Acceptance Report - Update: The First 43 Cycles Since Acceptance (November 29, 2000 to November 30, 2002),* April 2003, NASA/TM-2001-209984/Volume 4. [http://gfo.wff.nasa.gov/docs.html]
- D. W. Hancock III, G.S. Hayne,R.L. Brooks, D.W. Lockwood, *GFO Altimeter Engineering Assessment Report - From Launch to Acceptance (November 29, 2000 to November 21, 2001),* March 2002, NASA/TM-2001-209984/Volume 3. [http://gfo.wff.nasa.gov/docs.html]
- N. Tran, D.W. Hancock, G.S. Hayne, D. W. Lockwood, et al., *Assessment of the Cycle-Per-Cycle Noise Level of the Geosat Follow-On, TOPEX, and Poseidon Altimeters,* September 2001. [http://gfo.wff.nasa.gov/docs.html]
- M.L. Driscoll and R. V. Sailor, *GFO On-Orbit Altimeter Noise Assessment,* March 2001, NASA/TM-2001-209984/Volume 2. [http://gfo.wff.nasa.gov/docs.html]
- D.W. Hancock III, G.S. Hayne, R.L. Brooks, and D.W. Lockwood, *GFO Altimeter Engineering Assessment Report - From Launch to Acceptance (10 February 1998 to 29 November 2000),* March 2001, NASA/TM-2001-209984/Volume 1. [http://gfo.wff.nasa.gov/docs.html]
- GFO Cal/Val Data User's Handbook, July 1998, [http://gfo.bmpcoe.org/Gfo/Data_val/Cal_formats/formats.htm]
- *Interface Control Document for Payload Operations Center/ADFC for the Geosat Follow-On System,* December 1996, Ball Aerospace Systems Division, Doc. No. SP0025-617.
- G.S. Hayne, *Geosat Follow-On Altimeter Height Loop and AGC Loop Step Responses from Ground Testing,* April 1996. [http://gfo.wff.nasa.gov/docs.html]
- *Software Requirements Specification for the Geosat Follow-On (GFO) Radar Altimeter,* September 1994, E-Systems, Inc., Doc. No. 79-09-001-115(3).

Section 3

Radar Altimeter (RA) Files

3.1 Definition

The Radar Altimeter (RA) data consists of four types of product data files containing payload science and engineering data. The type of files are: 1) Altimeter File; 2) Calibration File; 3) Engineering File; and 4) Radiometer File. The formats for the various data products can be found at:

http://gfo.bmpcoe.org/GFO/Data_val/Cal_formats/formats.htm

3.1.1 Altimeter File Definition

The Altimeter file contains altimeter science data without waveforms and has the filename prefix of "ra_data". The "ra_data" file record is a short telemetry frame containing the raw RA data produced while the RA is in the standby or track mode.

3.1.2 Calibration File Definition

The Calibration file contains altimeter science data with waveforms and has the filename prefix of "ra_cal_data". The "ra_cal_data" file record is a long telemetry frame containing the raw RA data produced while the RA is in the Calibrate mode.

3.1.3 Engineering File Definition

The Engineering file contains a subset of the full ancillary engineering data set and has the filename prefix of "eng_data". The "eng_data" file record contains the data converted to engineering units.

3.1.4 Radiometer File Definition

The Radiometer file contains water vapor radiometer data and has the filename prefix of "wvr_data". The "wvr_data" file record contains the data converted to ASCII notation with no engineering unit conversion.

3.2 Distribution to Wallops

The daily near-real-time RA data is FTP'd from the Naval Oceanographic Office (NAVO), Altimetry Data Fusion Center (ADFC), Stennis Space Center, Bay St. Louis, MS to Wallops Flight Facility (WFF).

3.3 Distribution from Wallops

WFF forwards the "ra_cal_data" to other members of the Cal/Val Team.

3.4 Segment Processing

proc_gfo_ascra is the primary data reduction FORTRAN program. It is highly interactive, allowing the user to choose what process to run and to specify customized

parameters for the chosen process. Table 3-1 indicates the type of processing and output files that are available.

Table 3-1 RA Processing Modules and Output Files

Type Data Processing	Input Files Needed	Output Files/Suffix
Altimeter Data (short telemetry frame)	ra_data eng_data	.sciavg/Average Science (Appendix D.1) .engavg/Average Engineering (Appendix D.2) .log/Log (Appendix B.1)
Calibration Data (Long Telemetry Frame)	ra_cal_data eng_data	.wfcavg/Average Calibration WF (Appendix D.3) .wffavg/Average Fine Track WF (Appendix D.3) .engavg/Average Engineering (Appendix D.2) .log/Log (Appendix B.1)
Engineering Data	eng_data	.engavg/Average Engineering (Appendix D.2) .log/Log (Appendix B.1)

3.5 Daily Processing

autogfodailyra is a UNIX script file that runs the FORTRAN program, **proc_gfo_ascra**, and UNIX utilities, to process a day's worth of RA data segments. Refer to Appendix H.1, Script for Processing Daily RA Data, for the script listing. Figure 3-1 provides a diagram of the daily processing flow of the RA data. Figure 3-2 is an example of the processing of an RA data segment.

3.6 Special Processing

autoquicklookra is a UNIX script file that runs the FORTRAN program, **proc_gfo_ascra**; IDL program, **gfosciquicklook**; and UNIX utilities, to process a RA data segment immediately upon receipt at WFF. See Appendix H.8, Script for Quick Look RA Process, for the script listing.

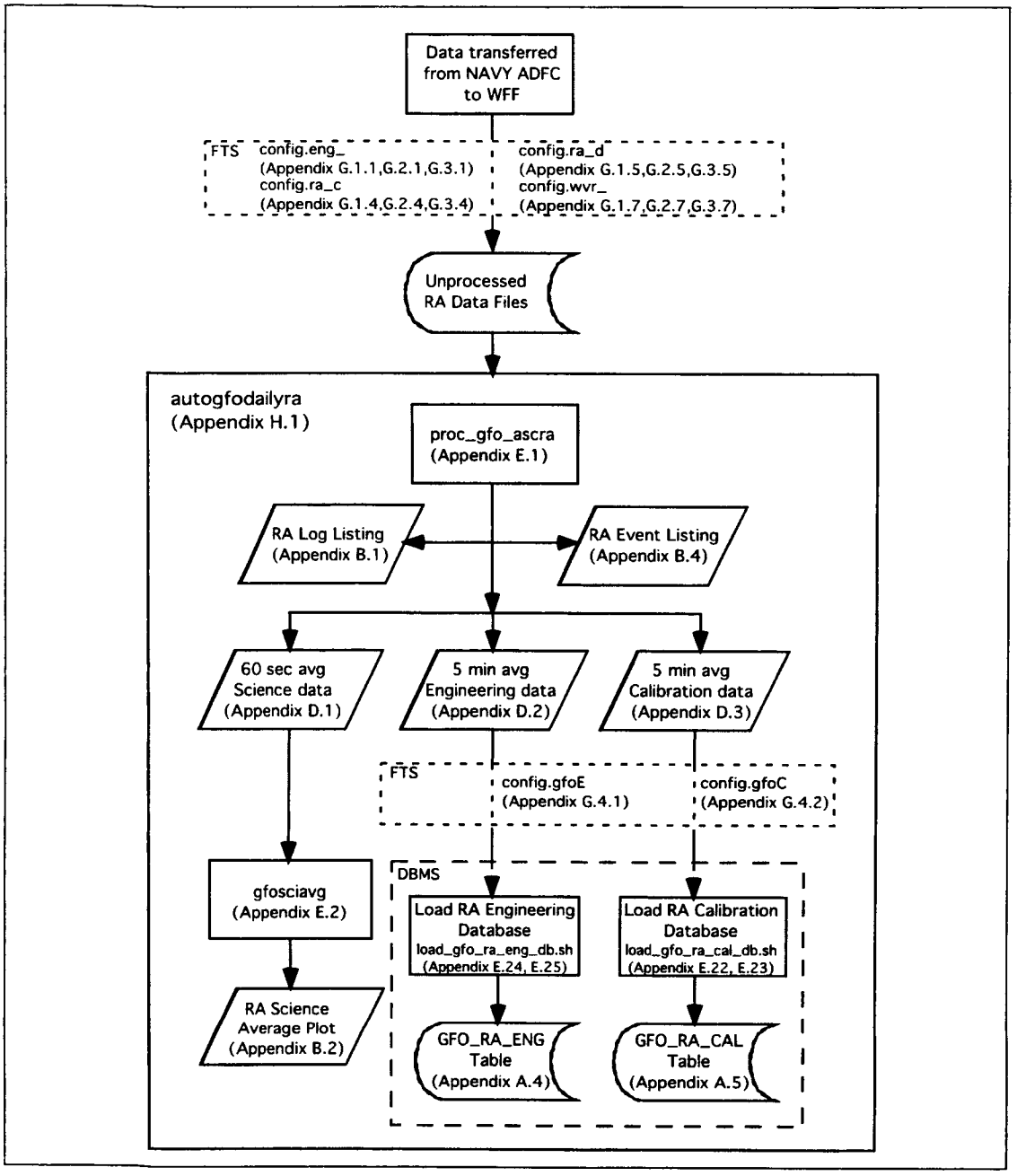

Figure 3-1 RA Daily Processing Data Flow

```
proc_gfo_ascra
 Initializations...

============================================================
                      W F F   G F O
            S O F T W A R E  D E V E L O P M E N T  T E A M
============================================================

          Program Number  : proc_gfo_ascra
          Program Title   : GFO ASC RA File Processing
          Program Version : Version 1.0 07/21/97

          Date of Run     : 05/05/03


 Enter type of Cal/Val File requested:
   0. Exit
   1. Altimeter File - Short Telemetry Frame
   2. Calibration File - Long Telemetry Frame
   3. Engineering File Only
   ?. Water Vapor Radiometer File
2

 Enter the complete GFO Data Filename
ra_cal_data03125_02_09_12.dat
 SciFileUTC 2003-125T02:09:12.000000

------------------------------------------------------------------------------
                           Select Processing
                     RETURN = Standard Processing
------------------------------------------------------------------------------

                      (0) Standard Processing.
                      (1) Average Science Data.
                      (2) Average Engineering Data.
                      (3) Average Calibration Waveforms
                      (4) Average Fine Track Waveforms
                      (5) Ouput Ascii Cal Data only.
                      (x) Exit.
3
 Process Selected : Average Calibration Data.

 Defaults : Seconds to Average:  10
          : Process CAL-1 and CAL-2.
```

Figure 3-2 Example of Processing RA Data

```
------------------------------------------------------------------------
                          Use Default Settings ?
                              (RETURN = Yes)
------------------------------------------------------------------------

                       1 = Yes, use default parameters
                       2 = No, set custom parameters
                       X = Exit.
1

Using Default Parameters.

  Opened Log File.....................gfo_raL_03125_02_09_12.log
  Opened Waveform CAL Averages File......gfo_raL_03125_02_09_12.wfcavg
  Opened Events File.................gfo_raL_03125_02_09_12.event


Processing...

First Sci frame number ;    0
First Sci frame seconds:      7752.4193261400
First Sci frame UTC     : 2003-125T02:09:12.419326
First Eng frame number ;    0
First Eng frame seconds:      7752.6935687100
First Eng frame UTC     : 2003-125T02:09:12.693569


Final Sci frame number ;    123953
Final Sci frame seconds:      19898.839297890
Final Sci frame UTC     : 2003-125T05:31:38.839298
Final Eng frame number ;    5929
Final Eng frame seconds:      19895.076746450
Final Eng frame UTC     : 2003-125T05:31:35.076746

------------------------------------------------------------------------
                          Closing All Open Files

Closing File : ra_cal_data03125_02_09_12.dat
Closing File : eng_data03125_02_09_12.dat
Closing File : gfo_raL_03125_02_09_12.log
Closing File : gfo_raL_03125_02_09_12.wfcavg
Closing File : gfo_raL_03125_02_09_12.event


Processing Complete.
```

Figure 3-2 Example of Processing RA Data (Continued)

```
============================================================
                        W F F   G F O
              S O F T W A R E   D E V E L O P M E N T   T E A M
============================================================

          Program Number  : proc_gfo_ascra
          Program Title   : GFO ASC RA File Processing
          Program Version : Version 1.0 07/21/97

          Date of Run     : 05/05/03


 Enter type of Cal/Val File requested:
   0. Exit
   1. Altimeter File - Short Telemetry Frame
   2. Calibration File - Long Telemetry Frame
   3. Engineering File Only
   ?. Water Vapor Radiometer File
0

 Program Ends.
```

Note: Manual entrys are indicated in **bold** type.

Figure 3-2 Example of Processing RA Data (Continued)

3.7 Trend Processing

Averaged GFO engineering and science parameters are loaded daily into database tables at WFF, permitting the user to retrieve present and past values. These values are entered into time-sequenced plots, useful for discerning any short-term (e.g., seasonal) or long-term trends. Figure 3-3 provides a diagram of the trend processing of the RA data.

gfohgtagccor is the UNIX script file that runs an IDL program, **gfohgtagccor.pro**. This program provides a plot of Cal1 Height and the AGC for Cal1 and Cal2. See Appendix B.15 for an example of the Calibration Trend plot.

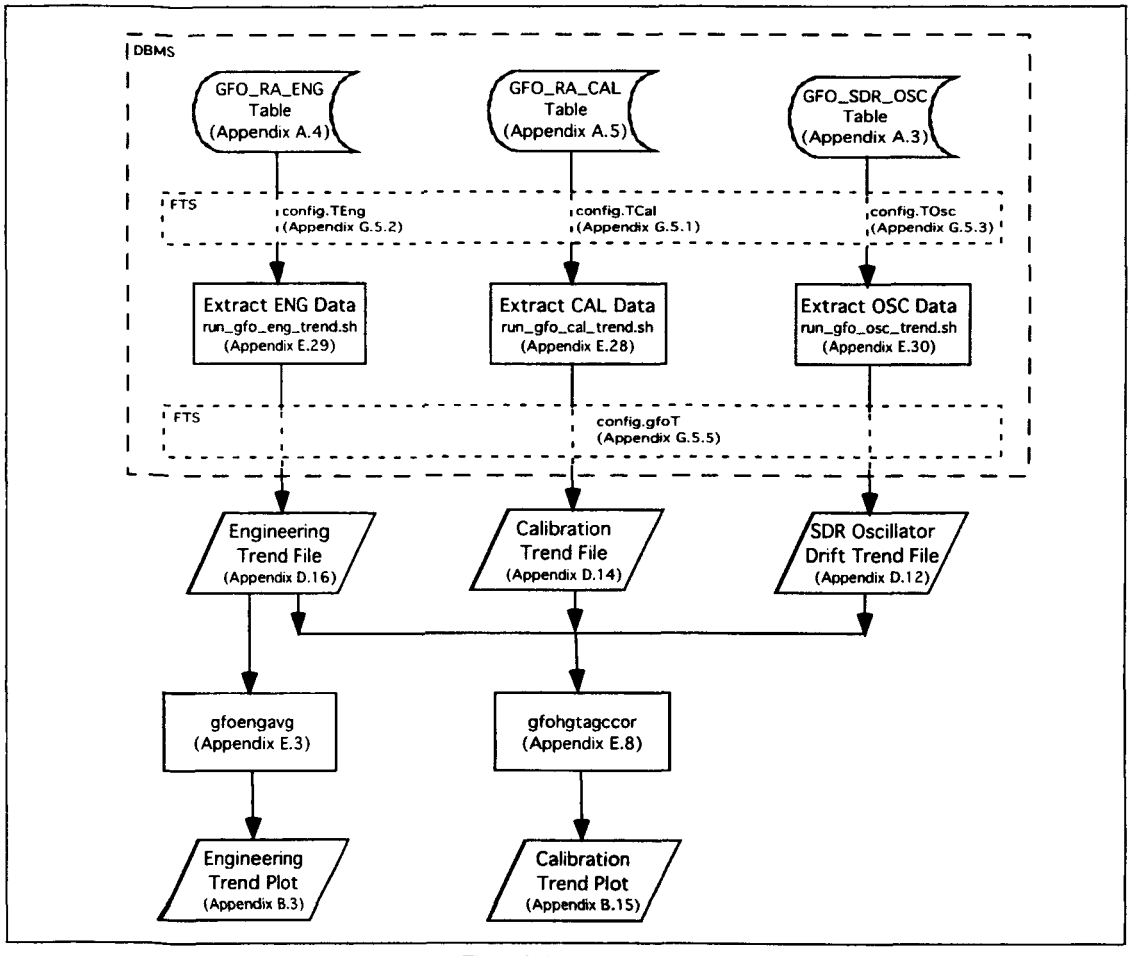

Figure 3-3 Trend Processing Data Flow

3.8 Dump Process

DumpGFO, Appendix E.18, is a FORTRAN program that will produce an output dump file for any of the Cal/Val, Sensor, or Geophysical data records. Appendix D.4, D.5, and D.6 provide the output formats for the Cal/Val data (ra_cal_data; ra_data; eng_data). Figure 3-4 is an example of the processing of an RA_CAL data segment.

```
DumpGFO
 Select Type Data File to Dump
   0 - Exit
   1 - ra_data (Short)
   2 - ra_cal_data (Long)
   3 - ra_eng
   4 - sdr
   5 - gdr
2
 Enter GFO RA Science(Long) Filename
ra_cal_data03125_02_09_12.dat

--------------------------------------------------------------------------------
                              Select Data Range
                            (RETURN = Process All)
--------------------------------------------------------------------------------

                    1 = Process All Data
                    2 = Process Selected Start & Stop Times
                    X = Exit
1

 Processing All Data.
 Select Type Data File to Dump
   0 - Exit
   1 - ra_data (Short)
   2 - ra_cal_data (Long)
   3 - ra_eng
   4 - sdr
   5 - gdr
0
 Program Ends
```

Figure 3-4 Example of Dump RA_CAL Data

Section 4

Sensor Data Record (SDR) Files

4.1 Definition

The Sensor Data Record (SDR) files contains altimeter and radiometer data when the altimeter is in fine track mode; it has the filename prefix of "sdr".

4.2 Distribution to Wallops

The daily near-real time SDR data is transferred from the Naval Oceanographic Office (NAVO), Altimetry Data Fusion Center (ADFC), Stennis Space Center, Bay St. Louis, MS to Wallops Flight Facility (WFF).

4.3 Distribution from Wallops

WFF forwards the "sdr" to other members of the Cal/Val Team.

4.4 Segment Processing

proc_gfo_sdr is the data reduction FORTRAN program. It is highly interactive, allowing the user to choose what process to run and to specify customized parameters for the chosen process.

Table 4-1 indicates the type of processing and output files that are available.

Table 4-1 SDR Processing Modules and Output Files

Type Data Processing	Input Files Needed	Output Files/Suffix
SDR Data	sdr	.sciavg/Average Science (Appendix D.11) .osc/Oscillator Drift (Appendix D.12) .log/Log Listing (Appendix B.5) .err/Error Count (Appendix D.13)

4.5 Daily Processing

autogfodailysdr is the UNIX script file that runs the FORTRAN program, **proc_gfo_sdr**; IDL program, **gfosciavg**; and UNIX utilities, to process a day's worth of SDR data segments. Refer to Appendix H.2, Script for Processing Daily SDR Data, for the script listing. Figure 4-1 provides a diagram of the daily processing flow of the SDR data. Figure 4-2 is an example of the processing of an SDR data segment.

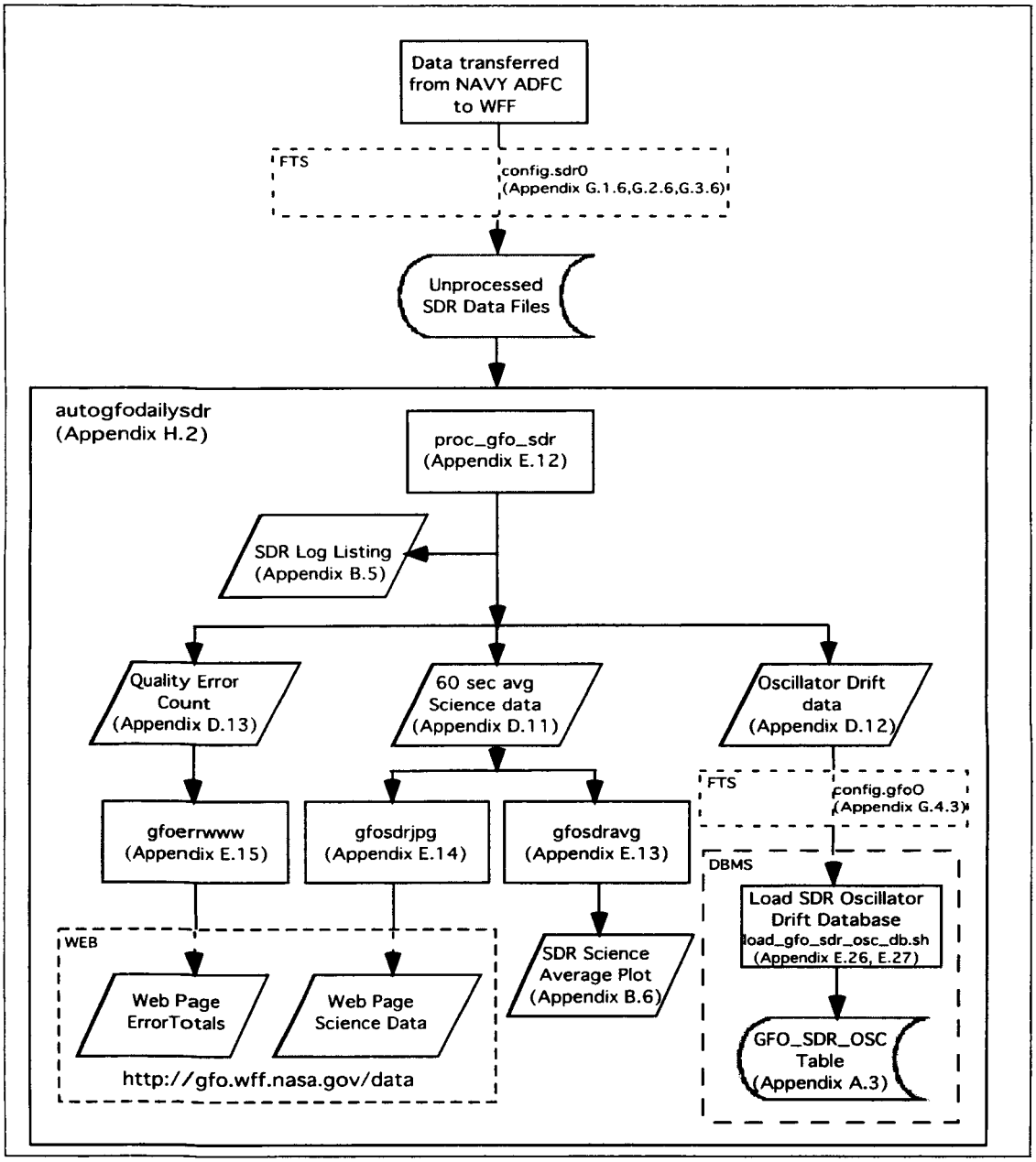

Figure 4-1 SDR Daily Processing Data Flow

```
proc_gfo_sdr
 Enter 0 to Exit  or  1 to Continue
1
 Enter GFO Sensor Data (SDR) Filename
sdr03125_02_09_13_12395.dat
 Enter Rate to Average
  -1 = 60 sec avg for db
   0 = Full Rate(10persec
       Only HgtWord,SWH,AGC are 10persec
   1 = 1 sec avg
   2 = 2 sec avg
    etc

1
 Select "Not in Fine Track" or "Fine Track Only"
 Enter 0 = Not In Fine Track and In Fine Track
       1 = In Fine Track Only

1

--------------------------------------------------------------------------------
                              Select Data Range
                            (RETURN = Process All)
--------------------------------------------------------------------------------

                    1 = Process All Data
                    2 = Process Selected Start & Stop Times
                    X = Exit
1

 Processing All Data.
     Opened Science Average File.....gfo_sdr03125_02_09_13_12395.sciavg
     Opened Event File.....gfo_sdr03125_02_09_13_12395.event
     Opened Log File.....gfo_sdr03125_02_09_13_12395.log
     Opened Error File.....gfo_sdr03125_02_09_13_12395.err
     Opened Osc File.....gfo_sdr03125_02_09_13_12395.osc
     Closed Science Average File.....gfo_sdr03125_02_09_13_12395.sciavg
     Closed Event File.....gfo_sdr03125_02_09_13_12395.event
     Closed Log File.....gfo_sdr03125_02_09_13_12395.log
     Closed Error File.....gfo_sdr03125_02_09_13_12395.err
     Closed Osc File.....gfo_sdr03125_02_09_13_12395.osc


 SDR Processing Complete.

 Enter 0 to Exit  or  1 to Continue
0

 Program Ended.
```

Note: Manual entrys are indicated in **bold** type.

Figure 4-2 Example of Processing SDR Data

4.6 Special Processing

autoquicklooksdr is a UNIX script file that runs the FORTRAN program, **proc_gfo_sdr**; IDL program, **gfosdrquicklook**; and UNIX utilities, to process a sdr data segment immediately upon receipt at WFF. Appendix B.7 provides a product sample of the SDR quicklook plot. See Appendix H.4, Script for Quick Look SDR Process, for the script listing.

4.7 Dump Process

DumpGFO, Appendix E.18, is a FORTRAN program that will produce an output dump file for any of the Cal/Val, Sensor, or Geophysical data records. Appendix D.7 provides the output formats for the Sensor data (sdr_data). Figure 3-4 is an example of the processing of an RA_CAL data segment.

Section 5

Navy Geophysical Data Record (NGDR) Files

5.1 Definition

The Navy Geophysical Data Record (NGDR) files contain the SDR data with appended earth location and geophysical corrections and has the filename prefix of "ngdr".

5.2 Distribution to Wallops

The NGDR data is transferred from the Naval Oceanographic Office (NAVO), Altimetry Data Fusion Center (ADFC), Stennis Space Center, Bay St. Louis, MS to Wallops Flight Facility (WFF).

5.3 Daily Processing

proc_gfo_gdr is the primary data reduction FORTRAN program. It is highly interactive, allowing the user to choose what process to run and to specify customized parameters for the chosen process.

Table 5-1 indicates the type of processing and output files that are available.

Table 5-1 NGDR Processing Modules and Output Files

Type Data Processing	Input Files Needed	Output Files/Suffix
NGDR	ngdr_gfo	.hdrdb/Header (Appendix D.9) .scidb/Average Science (Appendix D.10)

autogfodailyngdr is the UNIX script file that runs the FORTRAN program, **proc_gfo_gdr**; IDL programs, **gfo_ngdrpass, gfosciavg;** and UNIX utilities, to process a day's worth of ngdr data segments. A data segment for ngdr data is nominally for a 24-hour period. Refer to Appendix H.3, Script for Processing Daily NGDR Data, for the script listing. Figure 5-1 provides a diagram of the daily processing flow of the NGDR data. Figure 5-2 is an example of the processing of the NGDR data.

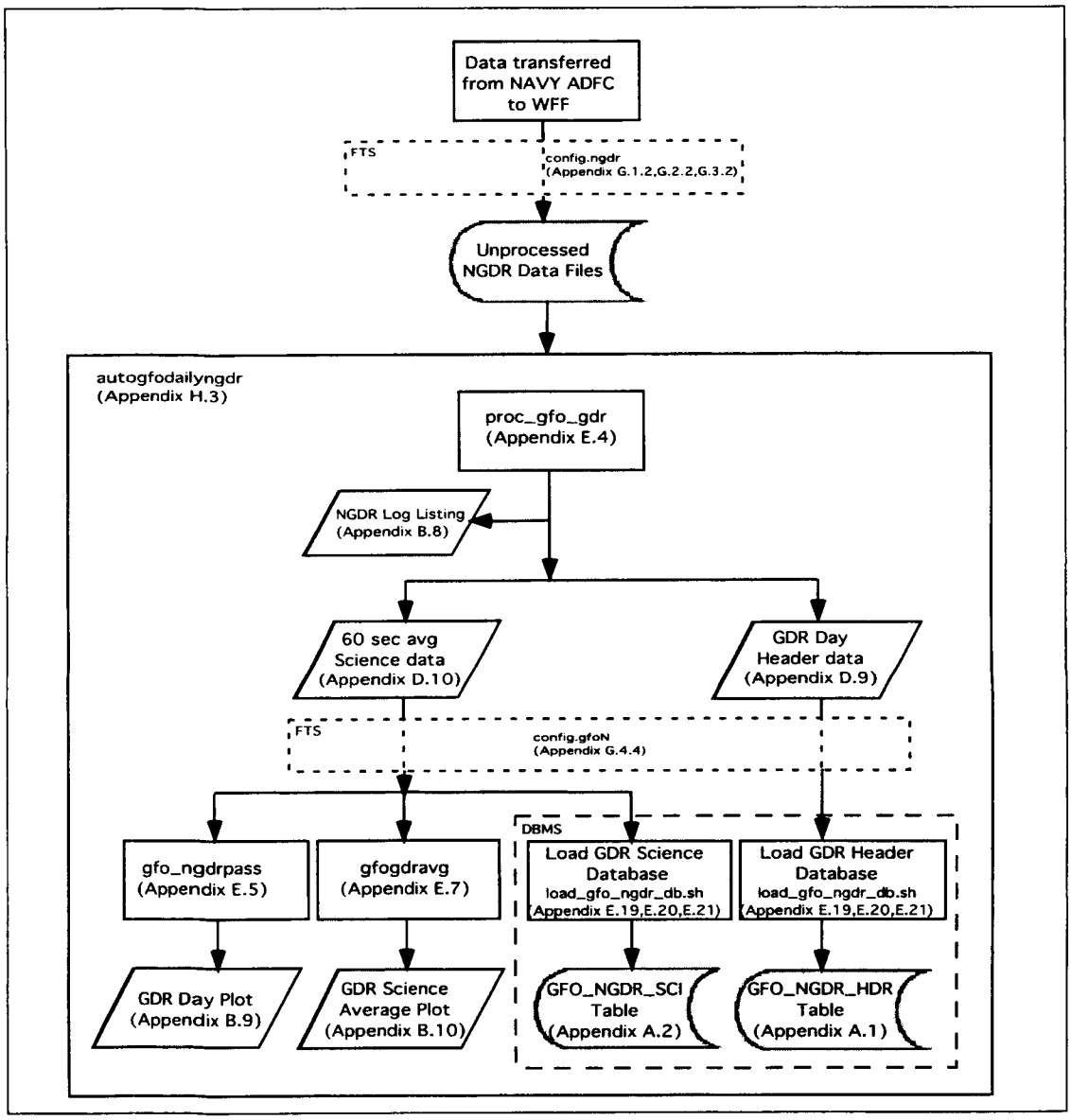

Figure 5-1 NGDR Daily Processing Data Flow

```
proc_gfo_gdr
Enter 0 to Exit  or  1 to Continue
1
 Enter GFO Sensor Data (GDR) Filename
ngdr_gfoo_2003125_00000_86399
 Enter Rate to Average
  -1 = 60 sec avg for db
   0 = Full Rate(10persec
       Only HgtWord,SWH,AGC are 10persec
   1 = 1 sec avg
   2 = 2 sec avg
    etc

1
 Select "Not in Fine Track" or "Fine Track Only"
 Enter 0 = Not In Fine Track and In Fine Track
       1 = In Fine Track Only

0

--------------------------------------------------------------------------------
                            Select Data Range
                          (RETURN = Process All)
--------------------------------------------------------------------------------

                    1 = Process All Data
                    2 = Process Selected Start & Stop Times
                    X = Exit
1

 Processing All Data.
     Opened Science Average File.......gfo_ngdr2003125_00000_86399.sciavg
     Opened Event File.................gfo_ngdr2003125_00000_86399.event
     Opened Log File...................gfo_ngdr2003125_00000_86399.log

 Delta SciTime Gap   105370649.999880   105371263.430714      613.43083405495
   5970
 Delta SciTime Gap   105372551.047441   105372553.987205      2.9397640228271
   7285
 Delta SciTime Gap   105384699.133279   105384702.073044      2.9397649765015
   19680
 Delta SciTime Gap   105388050.464534   105388662.915412      612.45087790489
   23098
 Delta SciTime Gap   105407351.976759   105407353.936602      1.9598430395126
   42171
 Delta SciTime Gap   105418938.567609   105418941.507373      2.9397640228271
   53994
 Delta SciTime Gap   105442980.939643   105442982.899486      1.9598429203033
   78527
```

Figure 5-2 Example of Processing NGDR Data

```
DO creategdrhdrrec   999  57864
     Closed Science Average File.......gfo_ngdr2003125_00000_86399.sciavg
     Closed Event File.................gfo_ngdr2003125_00000_86399.event
     Closed Log File...................gfo_ngdr2003125_00000_86399.log


 NGDR Processing Complete.

 Enter 0 to Exit  or  1 to Continue
0

 Program Ended.
```

Note: Manual Entrys are indicated in **bold** type.

Figure 5-2 Example of Processing NGDR Data (Continued)

5.4 Per-Cycle Processing

At the completion of each exact 17-day GFO cycle, the following processes are performed to produce per-cycle statistics and/or plots.

5.4.1 Process GFO Cycle Science Averages

The GEOSAT Follow-On (GFO) altimeter cycle averages.

- Change to working directory /gen/gfo/data/trend
  Example: cd /gen/gfo/data/trend
- Edit the GDR Cycle request, **CGDR_GDRCycle.req**, with desired cycle, 053
  Example: ed CGDR_GDRCycle.req
- Copy the GDR Cycle request to the directory /gen/gfo/dist/wff/in/ for GFO Database retrieval
  Example: cp CGDR_GDRCycle.req /gen/gfo/dist/wff/in
- The selected cycle file, gfoSycle053.gdr, will be placed in the /gen/gfo/data/trend/GDRSycleSum directory upon retrieval.
- Execute **gfogdrhist**, a UNIX script that runs an IDL program, **gfogdrhist.pro**. This program provides a plot of several parameters and an output Cycle Summary file, gfoSycle053.gdr.sum. See Appendix B.11 for an example of the History Cycle Plot.
  Example: gfogdrhist gfoSycle053.gdr
- Concatenate the Cycle Summary output file gfoSycle053.gdr.sum to gfocyclesummary.trend.
  Example: cat gfoSycle053.gdr.sum >> gfocyclesummary.trend
- Update GFOCycleSummary excel spreadsheet on the xserver located in Projects/GFO/Miscellaneous, using data from gfogdrhist plot or file gfoSycle053.gdr.sum in /gen/gfo/data/trend/GDRSycleSum.
- Print GFOCycleSummary spreadsheet
- Update web page – edit CycleTrend.html with new stats located in directory /gen/gfo/data/www.

5.4.2 Process GFO Range Measurement Noise

The GEOSAT Follow-On (GFO) altimeter white noise levels are evaluated using a technique based on high-pass filtering of 1-hz sea surface height time series. High-pass filtering removes the geoid and oceanography signals while revealing the random noise.

- Create a 17-day cycle directory in /raid/gfo
  Example: mkdir ngdr01207-01223.C14

- Copy the ngdr_gfoM data for the days of the cycle to the newly created cycle directory. The ngdr_gfoM data is normally stored in the /gen/gfo/store directory.
  Example:
  cp ngdr_gfoM_2001207_00001_86400 /raid/gfo/ngdr01207-01223.C14
  cp ngdr_gfoM_2001208_00001_86400 /raid/gfo/ngdr01207-01223.C14
  : :
  cp ngdr_gfoM_2001222_00001_86400 /raid/gfo/ngdr01207-01223.C14
  cp ngdr_gfoM_2001223_00001_86400 /raid/gfo/ngdr01207-01223.C14
- Change to the newly created cycle directory.
  Example: cd /raid/gfo/ngdr01207-01223.C14
- Execute **gfoallngdrdmp** – is a UNIX script that expands/compresses (gunzip/gzip) the ngdr data files and runs a FORTRAN program **DumpGFO**. This program will create a 1 second dump file for each ngdr available. See Appendix H.9, Script for NGDR Dump Process, for a listing of the script.
- Execute **do_highpassfilter_gfo_1min** – is a UNIX script that runs a FORTRAN90 program **highpassfilter_gfo_1min.f** and compresses (gzip) the ngdr data files. This program creates an output file OUT_do_highpassfilter_gfo_1min. See Appendix H.7, Script for Range Measurement Noise Process, for listing of script.
- Execute **do_stats_gfo_1min** – is a FORTRAN90 program **do_stats_gfo_1min.f** that calculates statistics from the file OUT_do_highpassfilter_gfo_1min and creates an output file **OUT_do_stats_gfo_1min**. There is manual input to this program that is based on the "Year, cycle number, first day of the data and last day of the data".
- Concatenate the statistics output file **OUT_do_stats_gfo_1min** to the summary file **do_stats_gfo_1min_summary.dat** with the previous cycle statistics.
  Example: cat OUT_do_stats_gfo_1min >> do_stats_gfo_1min_summary.dat
- Update web page – edit NoiseLevel.html with new stats located in directory /gen/gfo/data/www.

5.4.3 Process GFO-NCEP Wind Comparison

To co-locate and compute the statistical indicators for each cycle.

- In the directory /gen/ncep/COLOC_GFO_NCEP, create the directory for the new cycle.
  Example: mkdir coloc.C88, for cycle 88

- In the directory /gen/ncep/COLOC, edit script **do_coloc_gfo_ncep_cycle** to change file name & cycle number.
  Set DIR_GFO_INPUT = /raid/gfo/ngdr99999-99999.C88
  Set DIR_NCEP_OUT = /gen/ncep/COLOC_GFO_NCEP/coloc.C88
- Execute script **do_coloc_gfo_ncep_cycle**, this script collocates the NCEP data at each GFO locations, one file per day. See Appendix H.5, Script for Per-Cycle GFO-NCEP Co-locate Process, for listing of script.
- In the directory /gen/ncep/COLOC, edit script **do_avg_and_stat_data** to change file name & cycle number.
  Set DIR_GFO_INPUT = /raid/gfo/ngdr99999-99999.C88
  Set DIR_NCEP_OUT = /gen/ncep/COLOC_GFO_NCEP/coloc.C88
  Set YEAR = YYYY
  Set CYC = 88
- Execute script **do_avg_and_stat_data**, this script computes the 10 per second averaged data and the statistical indicators. The averaged data will be stored in the file, data_avg_YY_C88.dat, in the directory /gen/ncep/COLOC_GFO_NCEP/Data_filter_and_avg. The statistical indicators are stored in a summary file gfo_sigma0_summary.dat in the directory /gen/ncep/COLOC. See Appendix H.6, Script for Per-Cycle GFO-NCEP Averaging Process, for the script listing.

5.5 Cycle Summary Trend Processing

gfocycletrend is the UNIX script that runs an IDL program, **gfocycletrend.pro**. This program provides a trend plot of various parameters. See Appendix B.17 for an example of the Cycle Trend Plot.

5.6 Dump Process

DumpGFO, Appendix E.18, is a FORTRAN program that will produce an output dump file for any of the Cal/Val, Sensor, or Geophysical data records. Appendix D.8 provides the output formats for the Geophysical data (ngdr_data). Figure 3-4 is an example of the processing of an RA_CAL data segment.

Section 6

Operational Orbit Determination Data (OODD) Files

6.1 Definition

The operational orbit is a predicted orbit used primarily for mission planning and for quick-delivery data processing; it has the filename prefix of "oodd". Figure 6-1 provides a diagram of the flow of the OODD data.

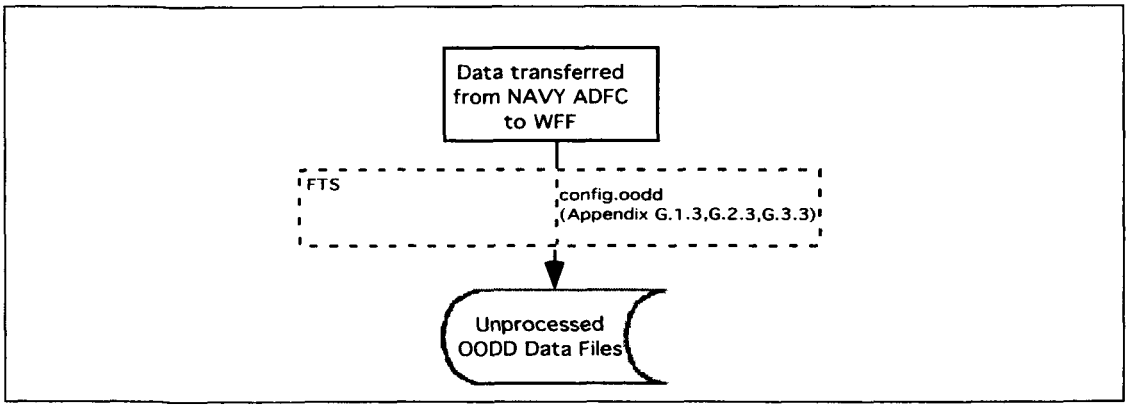

Figure 6-1 Flow of OODD Data

6.2 Distribution to Wallops

The OODD data is FTP'd from the Naval Oceanographic Office (NAVO), Altimetry Data Fusion Center (ADFC), Stennis Space Center, Bay St. Louis, MS to Wallops Flight Facility (WFF).

6.3 Distribution from Wallops

WFF forwards the "oodd" to other members of the Cal/Val Team.

Section 7

Directory Structure for GFO Processing

7.1 File System Directory Structure

The GFO processing and data storage are performed on the OSB3 workstation. The GFO Database Management is performed on the OSB8 workstation. The GFO file system is located on OSB3 with a main root directory of **/gen/gfo** with unique directories for data, source code, libraries, executables, etc.

Table 7-1 details the contents of each **/gen/gfo** directory. Figure 7-1 "Directory Structure" on page 7-3 provides a diagram for the main GFO file system.

In addition, there are two other root directories that are associated with the GFO file system. First, there is the directory of **/raid/gfo** that contains stored data for up to two years; Table 7-2 details the contents of each **/raid/gfo** directory. Second, there is the directory, **/gen/ncep**, that NCEP data and executables use in the GFO-NCEP wind comparison; Table 7-3 details the content of each **/gen/ncep** directory.

Table 7-1 OSB3:/gen/gfo Filesystem

Sub-directory	Contents
/bin	Executable binaries and script files
/data	Various GFO data directories
/incoming	
/ngdr	Temporary directory for processed ngdr data
/outgoing	
/quicklook	Temporary directory used for taking a quick look at incoming data
/ra	Temporary directory for RA data
/sdr	Temporary directory for processed sdr data
/trend	Temporary directory for trend data
/unprocessed	Temporary directory for unprocessed data
/www	
/dist	GFO Distribution and processing system files
/bin	Executable script files
/calval	Distribution directory for calval data being received
/config	Filetype configuration files
/in	Where all calval files to be processed originate
/out	Where calval processed files may be retrieved

Table 7-1 OSB3:/gen/gfo Filesystem (Continued)

/proc	Temporary holding area for files while they are being processed
/error	Where unknown files found in the "in" directory are placed
/log	Log entries for all GFO processing
/macart	Distribution directory for data being sent to John MacArthur
/config	Filetype configuration files
/in	Where all macart files to be processed originate
/out	Where macart processed files may be retrieved
/proc	Temporary holding area for files while they are being processed
/navo	Distribution directory for data being received from NAVO
/config	Filetype configuration files
/in	Where all navo files to be processed originate
/out	Where navo processed files may be retrieved
/proc	Temporary holding area for files while they are being processed
/wff	Distribution directory for data being processed at WFF
/bad	Rejected records from the database load
/config	Filetype configuration files
/dbtemp	Temporary holding area for processing group files
/in	Where all wff files to be processed originate
/out	Where wff processed files may be retrieved
/proc	Temporary holding area for files while they are being processed
/idl	IDL programs
/lib	Complied libraries
/src	Program F90 source codes
/store	6 month storage area for processed GFO data files
/wrk	Temporary working files

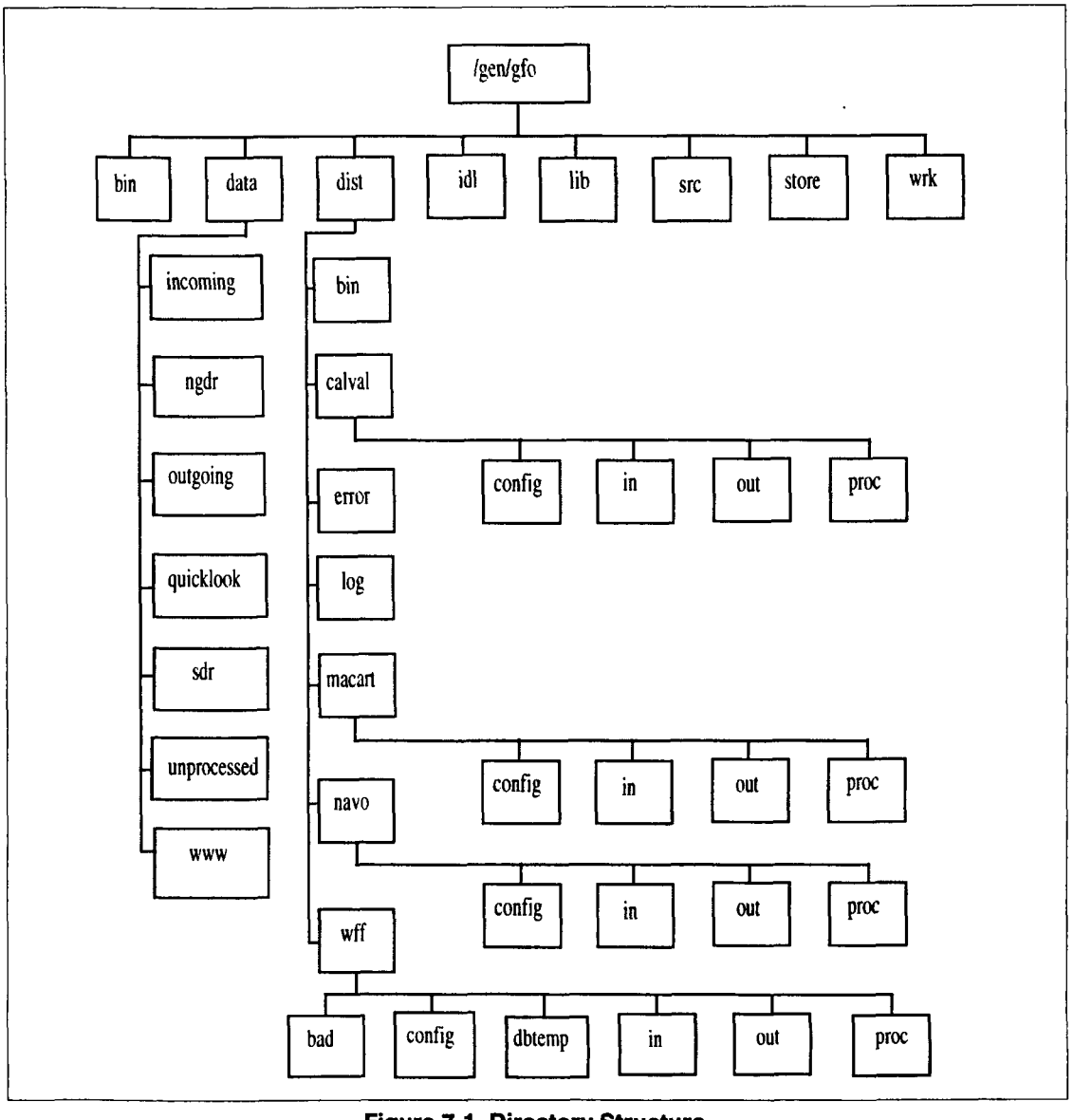

Figure 7-1 Directory Structure

Table 7-2 OSB3:/raid/gfo Filesystem

Sub-directory	Contents
/ngdr	Data storage for "ngdr" data only
/oodd	Data storage for "oodd" data only
/ra	Data storage for "ra" data only
/sdr	Data storage for "sdr" data only
/ngdrYYDOY.C88	GFO ngdr data for each 17-day cycle

Table 7-3 OSB3:/gen/ncep Filesystem

Sub-directory	Contents
/COLOC	
/COLOC_GFO_NCEP	
/Data_filter_and_avg	
/data_avg_YY_C88.dat	Average data files for each 17-day cycle
/coloc.C88	Co-location data for each 17-day cycle
/Data YYMM	NCEP data files for each year and month

Section 8

The File Transfer System (FTS)

8.1 Definition

The GFO File Transfer System (FTS) is designed to automate the handling of GFO CAL/VAL data with minimal user intervention. See Appendix F for the documentation and user's guide for the GFO FTS.

The GFO FTS was written using UNIX shell scripts and is based on several facilities of the UNIX operating system and the ORACLE database management system. The host OSB3 is the client system and the host OSB8 is the ORACLE server.

All files to be processed by the GFO FTS are placed in the "/gen/gfo/dist/*project*/in directory, where *project* represents either navo, calval, or wff. A process checks this directory every minute to see if a file has been placed in it. If so, the file is processed based on the existence of a filetype configuration (config.*filetype*) file in the "/gen/gfo/dist/*project*/config" directory. The config.*filetype* is a simple method of allowing the user to customize what happens to the files that are received, using the actions listed in the GFO FTS documentation in Appendix F.

There is a relationship between the filename being processed and the config.*filetype* name. The *filetype* is a unique 4-letter identifier based on the filename being processed. Table 8-1, shows this relationship.

Table 8-1 config.*filetype* vs. Filename to be Processed

config.**eng_**	**eng_data***yydoy_hh_mm_ss*.dat.gz
config.**ngdr**	**ngdr_gfoo***_yyyydoy_#####_#####*.gz
config.**oodd**	**oodd***yydoy_hh_mm_ss_####*.dat.gz
config.**ra_c**	**ra_cal_data***yydoy_hh_mm_ss*.dat.gz
config.**ra_d**	**ra_data***yydoy_hh_mm_ss*.dat.gz
config.**wvr_**	**wvr_data***yydoy_hh_mm.ss*.dat.gz
config.**sdr0**	**sdr0***ydoy_hh_ mm_ss*.dat.gz
config.**gfoE**	**gfoEng***yydoy*.db
config.**gfoC**	**gfoCal***yydoy*.db
config.**gfoO**	**gfoOsc***yydoy*.db
config.**gfoN**	**gfoNGDRhdr***yyyydoy*.db or **gfoNGDRsci***yyyydoy*.db
config.**TCal**	**TCal_**Caltrend.req

Table 8-1 config.*filetype* vs. Filename to be Processed (Continued)

config.**TEng**	**TEng**_Engtrend.req
config.**TOsc**	**TOsc**_OscTrend.req
config.**TGDR**	**TGDR**_gdrtrend.req
config.**gfoT**	**gfoT**rend.cal, **gfoT**rend.eng, **gfoT**rend.osc, **gfoT**rend.gdr
config.**CGDR**	**CGDR**_gdrcycle.req
config.**gfoS**	**gfoS**ycle###.gdr

See Appendix G for examples of the configuration files used for processing data that is placed in the "/gen/gfo/dist/*projects*/in" directory.

Section 9

Database Management System (DBMS)

The ORACLE Relational Database Management System (RDBMS) is being used to manage the GFO data. Using this system, GFO data can be loaded, extracted, searched, and sorted. Data is stored in database objects called tables.

9.1 Database Table Definitions

There are six database tables used in storing the GFO data.

- **GFO_NGDR_HDR** table contains header information for daily NGDR data records. (Appendix A.1)
- **GFO_NGDR_SCI** table contains 60-second-averaged science information for daily NGDR data records.(Appendix A.2)
- **GFO_RA_CAL** table contains twice-daily calibration data records. (Appendix A.5)
- **GFO_RA_ENG** table contains daily 10-second-averaged engineering data records. (Appendix A.4)
- **GFO_SDR_OSC** table contains daily SDR Oscillator drift data records. (Appendix A.3)
- **GFO_NGDR_SUM** table contains cycle summary averages. (Appendix A.6)

These database table definitions are documented in Appendix A.

9.2 Loading Data into the Database Tables

The GFO database tables are loaded using the Oracle utilities, **sqlldr**, **sqlplus**, and **PL/SQL**. The **sqlldr** utility loads data into the database tables using a control file, which maps the format of the input datafile to the database table. The **sqlplus** utility and **PL/SQL** procedures are used to perform miscellaneous checks and updates to database tables after loading. A log file, Appendix B.13, generated by the **sqlldr** utility is parsed and a smaller subset of information, Appendix B14, is created and printed. This log file shows the status of the data being loaded into the database. See Table 9-1 for specific config.*filetype* used for loading data into the database tables. Appendix I provides examples of Database Scripts for loading data.

Table 9-1 config.*filetype* vs. Database Management Tables

config.**gfoE**	Loading data into GFO_RA_ENG table
config.**gfoC**	Loading data into GFO_RA_CAL table
config.**gfoO**	Loading data into GFO_SDR_OSC table

Table 9-1 config.*filetype* vs. Database Management Tables (Continued)

config.**gfoN**	Loading data into GFO_NGDR_HDR table GFO_NGDR_SCI table
config.**TCal**	Extracting data from GFO_RA_CAL table
config.**TEng**	Extracting data from GFO_RA_ENG table
config.**TOsc**	Extracting data from GFO_SDR_OSC table
config.**TGDR**	Extracting data from GFO_NGDR_SUM table
config.**CGDR**	Extracting data from GFO_NGDR_HDR table GFO_NGDR_SCI table and loading data into GFO_NGDR_SUM table

9.3 Extracting Data from the Database Tables

The Oracle utilities, **sqlplus** and **PL/SQL**, are also used for extracting data from the database tables. These utilities are used to filter data and to create output files to be used in further processing. See Table 9-1 for specific config.*filetype* used for extracting data from the database tables. Appendix I provides examples of Database Scripts for extracting data.

The DBMS flowcharts for the extraction of trend plots for NGDR, RA, and SDR data are depicted in Figure 9-1, Figure 9-2, and Figure 9-3 respectively.

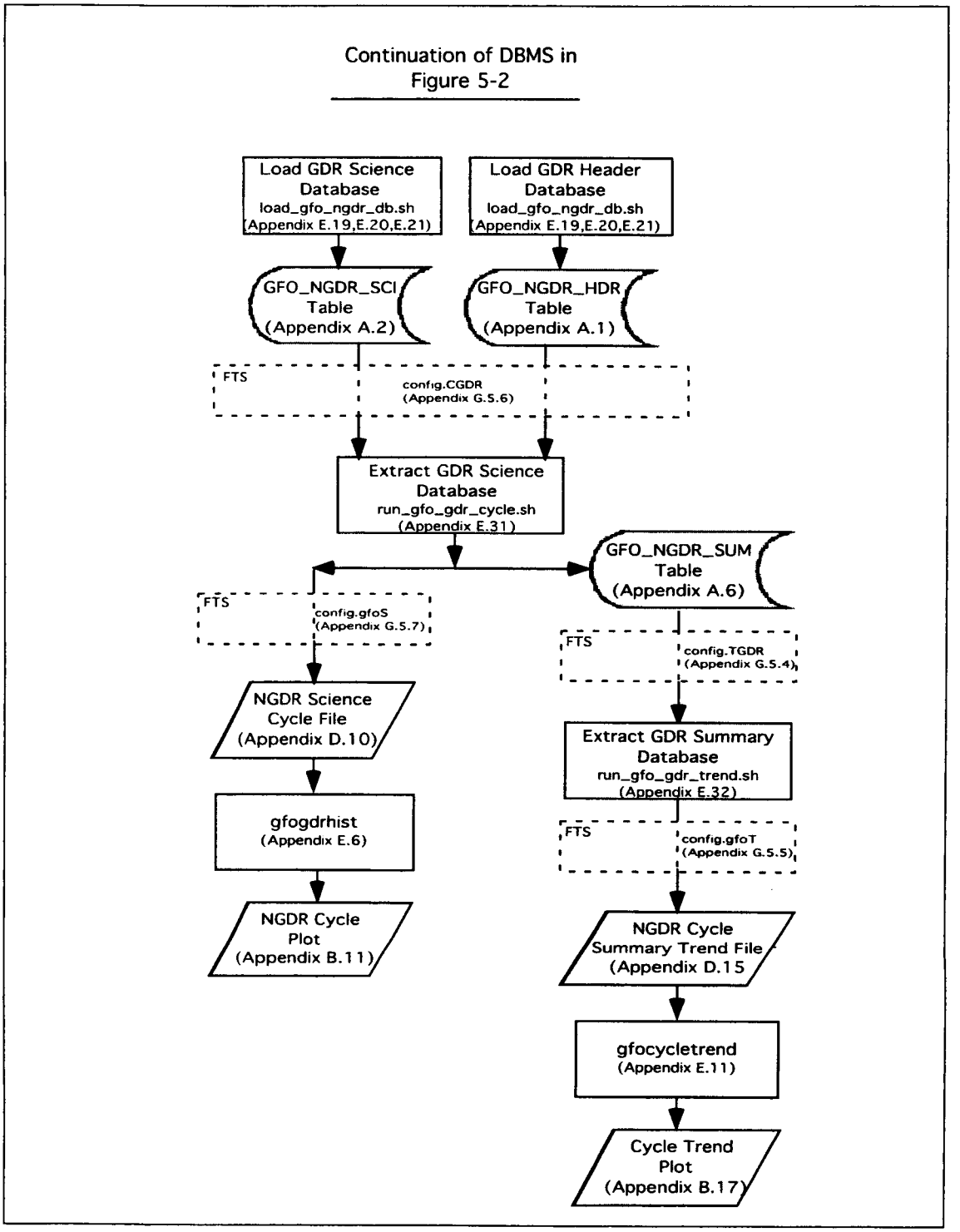

Figure 9-1 NGDR DBMS Data Flow

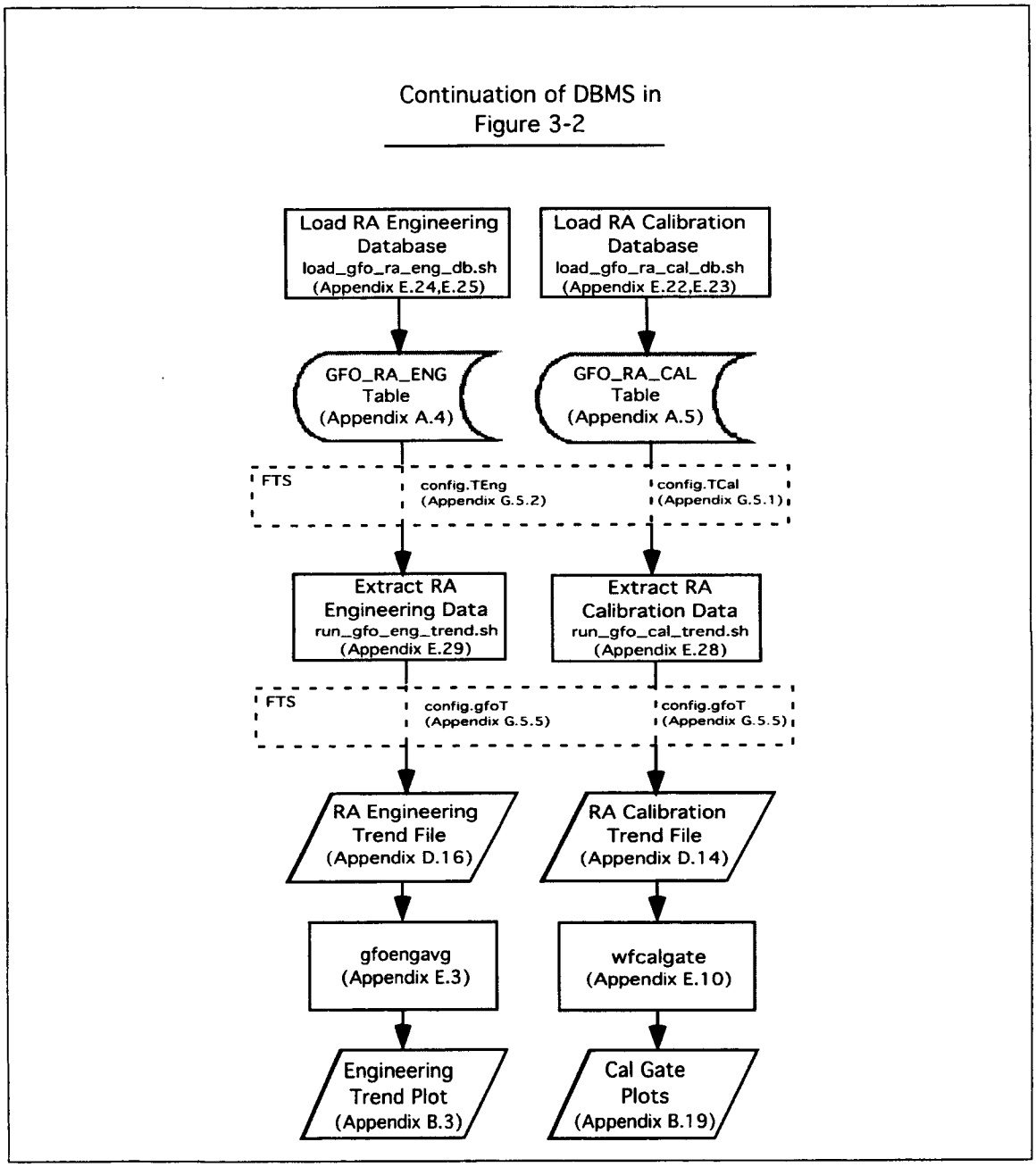

Figure 9-2 RA DBMS Data Flow

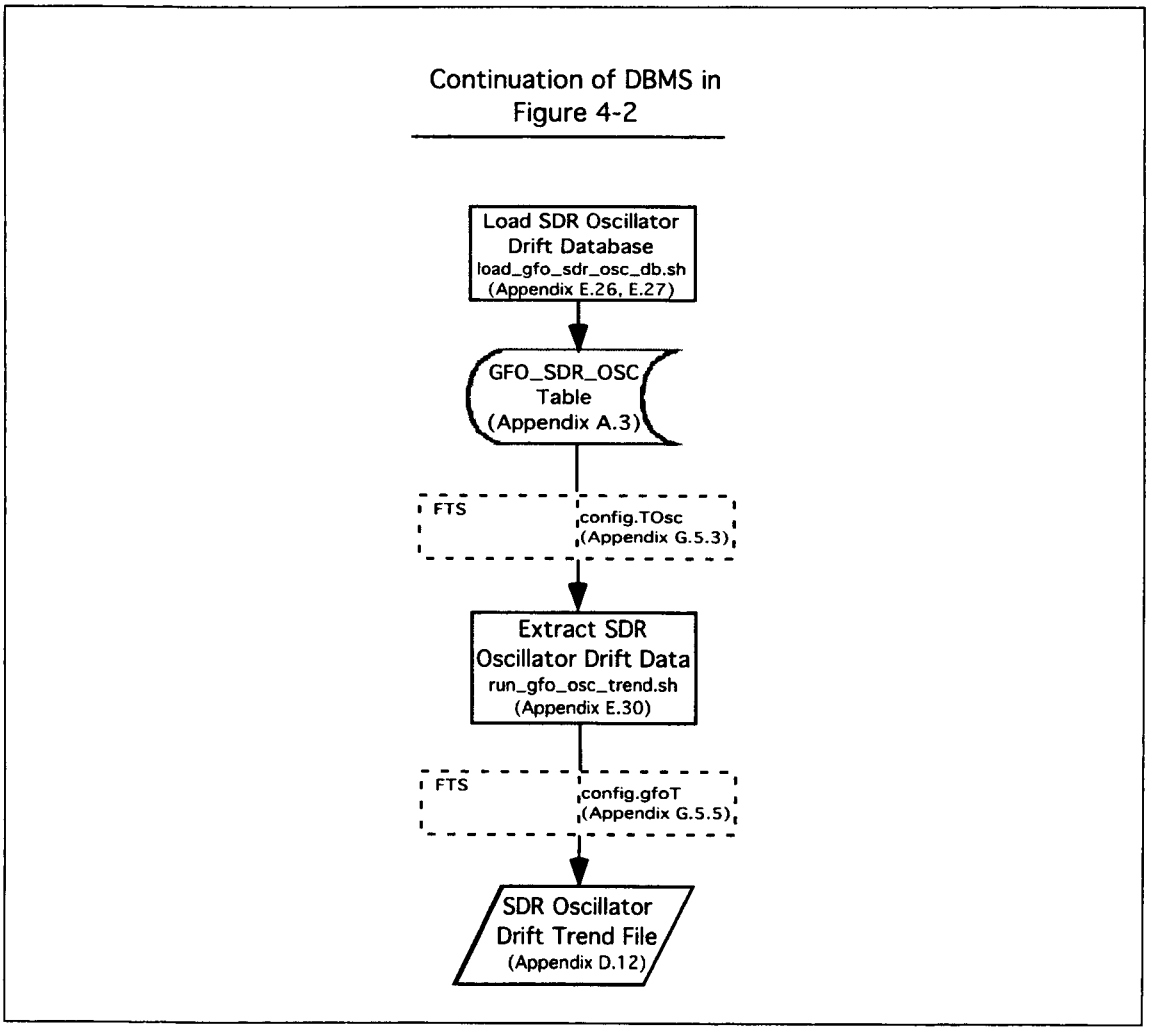

Figure 9-3 SDR DBMS Data Flow

Section 10

Cron Commanding

10.1 Definition

The cron (clock daemon) command initiates processes that execute at specified dates and times

GFO processing at Wallops Flight Facility involves three types of cron commands. They are utilized for: 1) daily processing; 2) watch inbox; 3) housekeeping. Figure 10-1 is an example of the cron table for GFO.

```
0  8 * * * csh /gen/gfo/bin/autogfodailyra
30 8 * * * csh /gen/gfo/bin/autogfodailyngdr
0 10 * * * csh /gen/gfo/bin/autogfodailysdr
* * * * * sh /gen/gfo/dist/bin/watch_inbox.sh calval 2>&1
* * * * * sh /gen/gfo/dist/bin/watch_inbox.sh wff 2>&1
* * * * * sh /gen/gfo/dist/bin/watch_inbox.sh navo 2>&1
* * * * * sh /gen/gfo/dist/bin/watch_inbox.sh macart 2>&1
0 0 * * * sh /gen/gfo/dist/bin/cleanout.sh all 2>&1
0 0 * * * sh /gen/gfo/dist/bin/newfiles.sh all 2>&1
```

Figure 10-1 Example of the GFO CRONTAB

Section 11

Points of Contact

GFO data is received from NAVO located at Stennis Space Center, MS. If there is a problem with the data being transferred to WFF, the following personnel may be contacted:

Lamar A. Russell
Email: russell@navo.navy.mil
Phone: (228)688-5684

Carolyn L. Cooper
Email: CooperCL@navo.navy.mil
Phone: (228)668-4935

Bruce Bricker
Email: bricker@navo.navy.mil
Phone: (228)688-4714

Their mail address is:
N321 – Synoptic Data Division
Real-Time Collection and Processing Branch
Naval Oceanographic Office
1002 Balch Blvd.
Stennis Space Center, MS 39522-5001

Computer Operations at NAVO perform a circuit check, "ping", to WFF every two hours to ensure circuit continuity. If there is no response after the second two-hour check, NAVO computer operations will notify WFF via phone line. Personnel notified at WFF are: 1) Dennis Lockwood, or 2) David Hancock.

Pertinent NAVO phone numbers are:

NAVOCEANO Computer Operations
Phone: (228)688-4402

NAVOCEANO Payload Operations Center (POC)
Phone: (228)688-4570

GFO Operations at NAVSOC that notify WFF concerning maneuvers and predicts are located at Point Mugu, CA. The contacts at Point Mugu are:

Marc Goldsmith
Email: goldsmm@satops.mugu.navy.mil
Phone: (805)488-3934

Dale Lao
Email: laod@satops.mugu.navy.mil
Phone: (805)488-3934

Their mail address is:
Satellite Engineer
General Dynamics
NAVSOC
Pt. Mugu, CA

The contacts regarding Central GFO Altimetry Mission concerns are:

GFO Project Manager at Computer Sciences Corporation
Morton Rau
Email: mrau@bmpcoe.org or mortrau@earthlink.net

Space Systems Consultant to Computer Sciences Corporation
Jay L. Finkelstein
Email: finkelsj@surffirst.net or finkelsj@compuserve.com

Appendix A
Database Table Definitions

A.1 NGDR Header Table Definition

Table A-1 NGDR Header Format

	Field	Units	Format	Description
	CYCLE	#	NUMBER(3)	Cycle = 17 days
	YRDOYR	#	VARCHAR2(7)	Year Day of Year
	FILETIME	char	VARCHAR2(11)	
	DATERUN	char	VARCHAR2(8)	Date of Process
	CHGTCALIBBIAS	mm	VARCHAR2(12)	Height Calibration Bias
	CALTBIASINIT	mm	VARCHAR2(12)	Attitude Bias Initial Correction
	CALTBIASGC	mm	VARCHAR2(12)	Attitude Bias Center of Gravity Correction
	CSWHBIASINIT	mm	VARCHAR2(12)	SWH Bias Initial
	CAGCCALIBBIAS	dB	VARCHAR2(12)	AGC Calibration Bias
	CAGCBIASINIT	dB	VARCHAR2(12)	AGC Bias Initial
	TOTFRAMES	cnts	NUMBER(6)	Total Number of Frames
	TOTUSED	cnts	NUMBER(6)	Total Number of Frames Used
	TOTDELETE	cnts	NUMBER(6)	Total Number of Frames Deleted
	TOTTRKFRAMES	cnts	NUMBER(6)	Total Number of Frames in Track
	KWORB	char	VARCHAR2(10)	Keyword value for Altitude
	KWTID	char	VARCHAR2(10)	Keyword value for Ocean Tide
	KWION	char	VARCHAR2(10)	Keyword value for Ionosphere
	KWDRY	char	VARCHAR2(10)	Keyword value for Dry Tropo-sphere
	KWWET	char	VARCHAR2(10)	Keyword value for Wet Tropo-sphere

A.2 NGDR Science Table Definition

Table A-2 NGDR Science Format

	Field	Units	Format	Description
	TIMESEC	sec	NUMBER(16,3)	Converted to 2000 Epoch
	ATB	date	VARCHAR2(24)	UTC time
	RECCOUNT	counts	NUMBER(5,1)	Number Frames Used in 60-sec Average
	TELEMFMT		VARCHAR2(5)	Telemetry Format (NORMS,FINEL)
	NOTINFTRK	#	NUMBER(4)	Value: 0=Not in FTRK, 1=In FTRK
	SSHCORRECTED	m	NUMBER(16,6)	SSHC = SSHU-Environment Corrections
	SSHUNCORRECTED	m	NUMBER(16,6)	SSHU=SatAtt-StRng+NetHgtCorr
	SSHUSTD	m	NUMBER(16,6)	Standard Deviation from Fit applied to SSHU Values
	WINDSPEED	cm/ sec	NUMBER(16,6)	Wind Speed from Sigma0
	SWH	m	NUMBER(16,6)	Significant Waveheight
	SWHSTD	m	NUMBER(16,6)	Standard Deviation from Fit applied to SWH Values
	SIGMA0	dB	NUMBER(16,6)	Backscatter Coefficient
	AGC	dB	NUMBER(16,6)	Automatic Gain Control
	AGCSTD	dB	NUMBER(16,6)	Standard Deviation from Fit applied to AGC Values
	DRYTOPO	m	NUMBER(16,6)	Dry Troposphere
	WETTROPO	m	NUMBER(16,6)	Wet Troposphere
	NETAGCCORR	dB	NUMBER(16,6)	Net AGC Correction
	ATTITUDE	deg	NUMBER(16,6)	Attitude (Off-Nadir Angle)
	IONO	m	NUMBER(16,6)	-1 * Att Rng Corr
	NETSWHCORR	m	NUMBER(16,6)	Net SWH Correction
	INVBARO	m	NUMBER(16,6)	Inverse Barometer
	BTEMP22	deg K	NUMBER(16,6)	22 GHz Brightness Temp
	BTEMP37	deg K	NUMBER(16,6)	37 GHz Brightness Temp
	VATT	v	NUMBER(16,6)	Averaged VATT
	VATTFIT	v	NUMBER(16,6)	Fitted VATT
	RECVRTEMP	deg C	NUMBER(16,6)	Receiver Temperature

Table A-2 NGDR Science Format (Continued)

	Field	Units	Format	Description
	LATITUDE	deg	NUMBER(16,6)	Geodetic Latitude +90N to -90S
	LONGITUDE	deg	NUMBER(16,6)	East Geodetic Longitude 0 to 360
	EMBIAS	m	NUMBER(16,6)	Sea State Bias
	NETHGTCORR	m	NUMBER(16,6)	Net Height Correction
	EARTHTIDE	m	NUMBER(16,6)	Solid Earth Tide
	OCEANTIDE	m	NUMBER(16,6)	Ocean Water Tide
	LOADTIDE	m	NUMBER(16,6)	Ocean Load Tide
	POLETIDE	m	NUMBER(16,6)	Pole Tide
	WATERDEPTH	m	NUMBER(16,6)	Water Depth
	GEOID	m	NUMBER(16,6)	Geoid Height
	SUBMODE	m	NUMBER(16,6)	Submode = ((4*TrkMode(1)) + (2*TypeTrack(1))+ (AGCSource(1))+ (4*TrkMode(2))+ (2*TypeTrack(2))+ (AGCSource(2)))/2

A.3 SDR Oscillator Drift Table Definition

Table A-3 SDR Oscillator Drift Format

	Field	Units	Format	Description
	Segment	txt	VARCHAR2(30)	Segment ID
	StartUTC	sec	NUMBER(16,9)	Start UTC
	UTCYear	year	NUMBER(4)	UTC Year
	UTCDOY	day	NUMBER(3)	UTC Day of Year
	UTCSeconds	sec	NUMBER(16,9)	UTC Seconds
	VTCWValue	ticks	NUMBER(18,12)	VTCW Value
	Ratio	sec/tick	NUMBER(26,22)	Ratio

Note: This data is extracted from the SDR Header

A.4 RA Engineering Table Definition

Table A-4 RA Engineering Format

Name	Units	Fmt	Description/Computation
TimeSec	sec	NUMBER(16,3)	UTC Time in seconds from J2000
ATB	txt	VARCHAR2(20)	UTC Date and Time
Stat	txt	VARCHAR2(4)	Type of Statistic (Ex: Mean)
TestID	txt	VARCHAR2(20)	RA Configuration (Ex: RA 1 SSPA 1)
Segment	txt	VARCHAR2(20)	Segment ID (filename)
RecCount	count	NUMBER(5)	Number of Records used in Computing Average
EngTemp01	degC	NUMBER(9,4)	RA 1 Receiver Temp
EngTemp02	degC	NUMBER(9,4)	RA 1 TRS 1 Temp
EngTemp03	degC	NUMBER(9,4)	RA 1 TRS 2 Temp
EngTemp04	degC	NUMBER(9,4)	RA 1 Receiver 1 Temp
EngTemp05	degC	NUMBER(9,4)	RA 1 Receiver 2 Temp
EngTemp06	degC	NUMBER(9,4)	RA 1 DFB Temp
EngTemp07	degC	NUMBER(9,4)	RA 2 Receiver Temp
EngTemp08	degC	NUMBER(9,4)	RA 2 TRS 1 Temp
EngTemp09	degC	NUMBER(9,4)	RA 2 TRS 2 Temp
EngTemp10	degC	NUMBER(9,4)	RA 2 Receiver 1 Temp
EngTemp11	degC	NUMBER(9,4)	RA 2 Receiver 2 Temp
EngTemp12	degC	NUMBER(9,4)	RA 2 DFB Temp
CompRecvrTemp	degC	NUMBER(9,4)	Composite Receiver Temp
CoLatitude	deg	NUMBER(9,4)	Co-Latitude(0-180)

A.5 RA Calibration Table Definition

Table A-5 RA Calibration Database Format

Name	Units	Fmt	Description/Computation
TimeSec	sec	NUMBER(16,4)	UTC Time in seconds from J2000
ATB	date	VARCHAR2(20)	UTC Date and Time
Latitude	id	NUMBER(13,6)	Latitude (-90 to +90)
Stat	id	VARCHAR2(4)	Min/Max/RMS/Mean/Full
TestID	txt	VARCHAR2(20)	RA Configuration (Ex: RA 1 SSPA 1)
Segment	txt	VARCHAR2(20)	Segment ID (filename)
RecCount	counts	NUMBER(5)	Number of records used in computing average
TrkFmt	txt	VARCHAR2(5)	Track Format (CAL1,CAL2)
Height	meters	NUMBER(18,8)	Height
AGC	dB	NUMBER(13,6)	AGC
SWH	m	NUMBER(13,6)	SWH
VATT	ratio	NUMBER(13,6)	VATT
AGCGateAmp	counts	NUMBER(13,6)	Scaled counts*(Waveform Scaling Factor+1)
LEBinCnt	counts	NUMBER(13,6)	Scaled counts*(Waveform Scaling Factor+1)
EarlyGateAmp	counts	NUMBER(13,6)	Scaled counts*(Waveform Scaling Factor+1)
LateGateAmp	counts	NUMBER(13,6)	Scaled counts*(Waveform Scaling Factor+1)
MidGateAmp	counts	NUMBER(13,6)	Scaled counts*(Waveform Scaling Factor+1)
AttGateAmp	counts	NUMBER(13,6)	Scaled counts*(Waveform Scaling Factor+1)
NoiseGateAmp	counts	NUMBER(13,6)	Scaled counts*(Waveform Scaling Factor+1)
SignalNoise	counts	NUMBER(13,6)	Scaled counts*(Waveform Scaling Factor+1)
SubMode	counts	NUMBER(13,6)	Scaled counts*(Waveform Scaling Factor+1)
GateIndex	counts	NUMBER(13,6)	Scaled counts*(Waveform Scaling Factor+1)
Gate01	counts	NUMBER(10,2)	Value for Gate 1
Gate02	counts	NUMBER(10,2)	Value for Gate 2
. . .	. . .	. . .	. . .
Gate63	counts	NUMBER(10,2)	Value for Gate 63
Gate64	counts	NUMBER(10,2)	Value for Gate 64

A.6 NGDR Cycle Summary Table Definition

Table A-6 NGDR Cycle Summary Format

	Field	Units	Format	Description
	TIMESEC	sec	NUMBER(16,3)	Beginning 2000 Epoch of Cycle
	ATB	date	VARCHAR2(24)	Beginning UTC Time of Cycle
	CYCLE	#	NUMBER(3)	Cycle Number
	STARTDATE	#	VARCHAR2(7)	Start Date of Cycle (YYYYDOY)
	STOPDATE	#	VARCHAR2(7)	Stop Date of Cycle (YYYYDOY)
	RECCOUNT	counts	NUMBER(12,1)	Number Frames Used in Cycle after Filter
	SSHCORRECTED	m	NUMBER(16,6)	Cycle Average (SSHC = SSHU-Environment Corrections)
	SSHUNCORRECTED	m	NUMBER(16,6)	Cycle Average (SSHU=SatAtt-StRng+NetHgtCorr)
	SSHUSTD	m	NUMBER(16,6)	Cycle Average (Standard Deviation from Fit applied to SSHU Values)
	WINDSPEED	cm/ sec	NUMBER(16,6)	Cycle Average (Wind Speed from Sigma0)
	SWH	m	NUMBER(16,6)	Cycle Average (Significant Waveheight)
	SWHSTD	m	NUMBER(16,6)	Cycle Average (Standard Deviation from Fit applied to SWH Values)
	SIGMA0	dB	NUMBER(16,6)	Cycle Average (Backscatter Coefficient)
	AGC	dB	NUMBER(16,6)	Cycle Average (Automatic Gain Control)
	AGCSTD	dB	NUMBER(16,6)	Cycle Average (Standard Deviation from Fit applied to AGC Values)
	DRYTOPO	m	NUMBER(16,6)	Cycle Average (Dry Troposphere)
	WETTROPO	m	NUMBER(16,6)	Cycle Average (Wet Troposphere)
	NETAGCCORR	dB	NUMBER(16,6)	Cycle Average (Net AGC Correction)
	ATTITUDE	deg	NUMBER(16,6)	Cycle Average (Attitude (Off-Nadir Angle))
	IONO	m	NUMBER(16,6)	Cycle Average (-1 * Att Rng Corr)
	NETSWHCORR	m	NUMBER(16,6)	Cycle Average (Net SWH Correction)
	INVBARO	m	NUMBER(16,6)	Cycle Average (Inverse Barometer)
	BTEMP22	deg K	NUMBER(16,6)	Cycle Average (22 GHz Brightness Temp)
	BTEMP37	deg K	NUMBER(16,6)	Cycle Average (37 GHz Brightness Temp)
	VATT	v	NUMBER(16,6)	Cycle Average (Averaged VATT)

Table A-6 NGDR Cycle Summary Format (Continued)

	Field	Units	Format	Description
	VATTFIT	v	NUMBER(16,6)	Cycle Average (Fitted VATT)
	RECVRTEMP	deg C	NUMBER(16,6)	Cycle Average (Receiver Temperature)
	EMBIAS	m	NUMBER(16,6)	Cycle Average (Sea State Bias)
	NETHGTCORR	m	NUMBER(16,6)	Cycle Average (Net Height Correction)
	EARTHTIDE	m	NUMBER(16,6)	Cycle Average (Solid Earth Tide)
	OCEANTIDE	m	NUMBER(16,6)	Cycle Average (Ocean Water Tide)
	LOADTIDE	m	NUMBER(16,6)	Cycle Average (Ocean Load Tide)
	POLETIDE	m	NUMBER(16,6)	Cycle Average (Pole Tide)
	WATERDEPTH	m	NUMBER(16,6)	Cycle Average (Water Depth)
	GEOID	m	NUMBER(16,6)	Cycle Average (Geoid Height)
	SUBMODE	m	NUMBER(16,6)	Cycle Average (Submode = ((4*TrkMode(1)) + (2*TypeTrack(1))+ (AGCSource(1))+ (4*TrkMode(2))+ (2*TypeTrack(2))+ (AGCSource(2)))/2)
	TOTFRAMES	counts	NUMBER(12,1)	Cycle Sum (Frames Available)
	TOTUSED	counts	NUMBER(12,1)	Cycle Sum (Frames Used)
	TOTDELETE	counts	NUMBER(12,1)	Cycle Sum (Frames Deleted)
	TOTTRKFRAMES	counts	NUMBER(12,1)	Cycle Sum (Frames in TRK)

Appendix B
Product Samples

B.1 RA Log Listing

```
gfo_raL_03125_02_09_12.log

--------------------------------------------------------------------------------
WFF VERSION : asc RA Software = Version 1.0 07/21/97
--------------------------------------------------------------------------------

First Sci frame number ;    0
First Sci frame seconds:      7752.4193261400
First Sci frame UTC     : 2003-125T02:09:12.419326
First Eng frame number ;    0
First Eng frame seconds:      7752.6935687100
First Eng frame UTC     : 2003-125T02:09:12.693569


Final Sci frame number ;    123953
Final Sci frame seconds:      19898.839297890
Final Sci frame UTC     : 2003-125T05:31:38.839298
Final Eng frame number ;    5929
Final Eng frame seconds:      19895.076746450
Final Eng frame UTC     : 2003-125T05:31:35.076746
```

Figure B-1 RA Log Listing

B.2 RA Science Average Plot

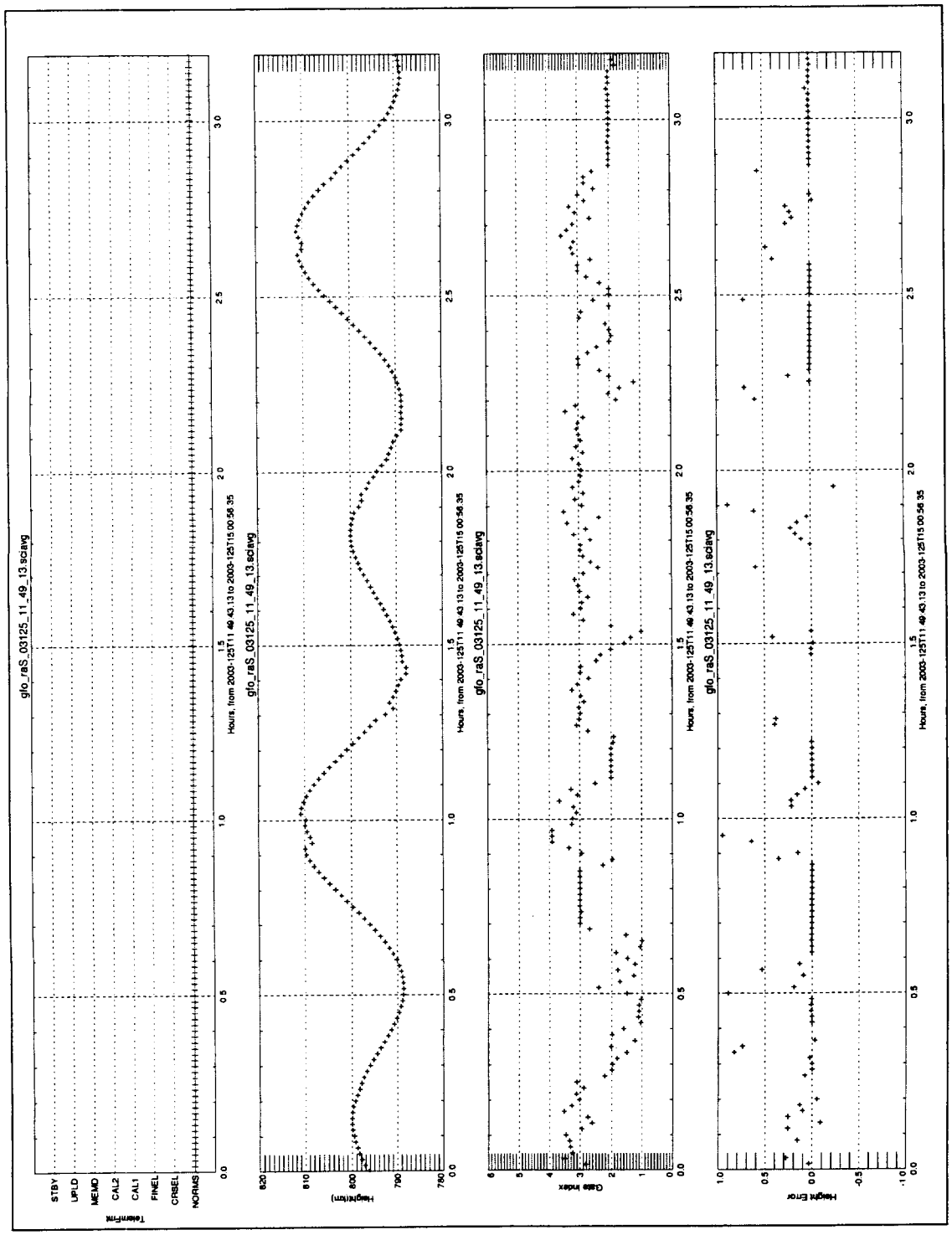

Figure B-2 RA Science Average Plot

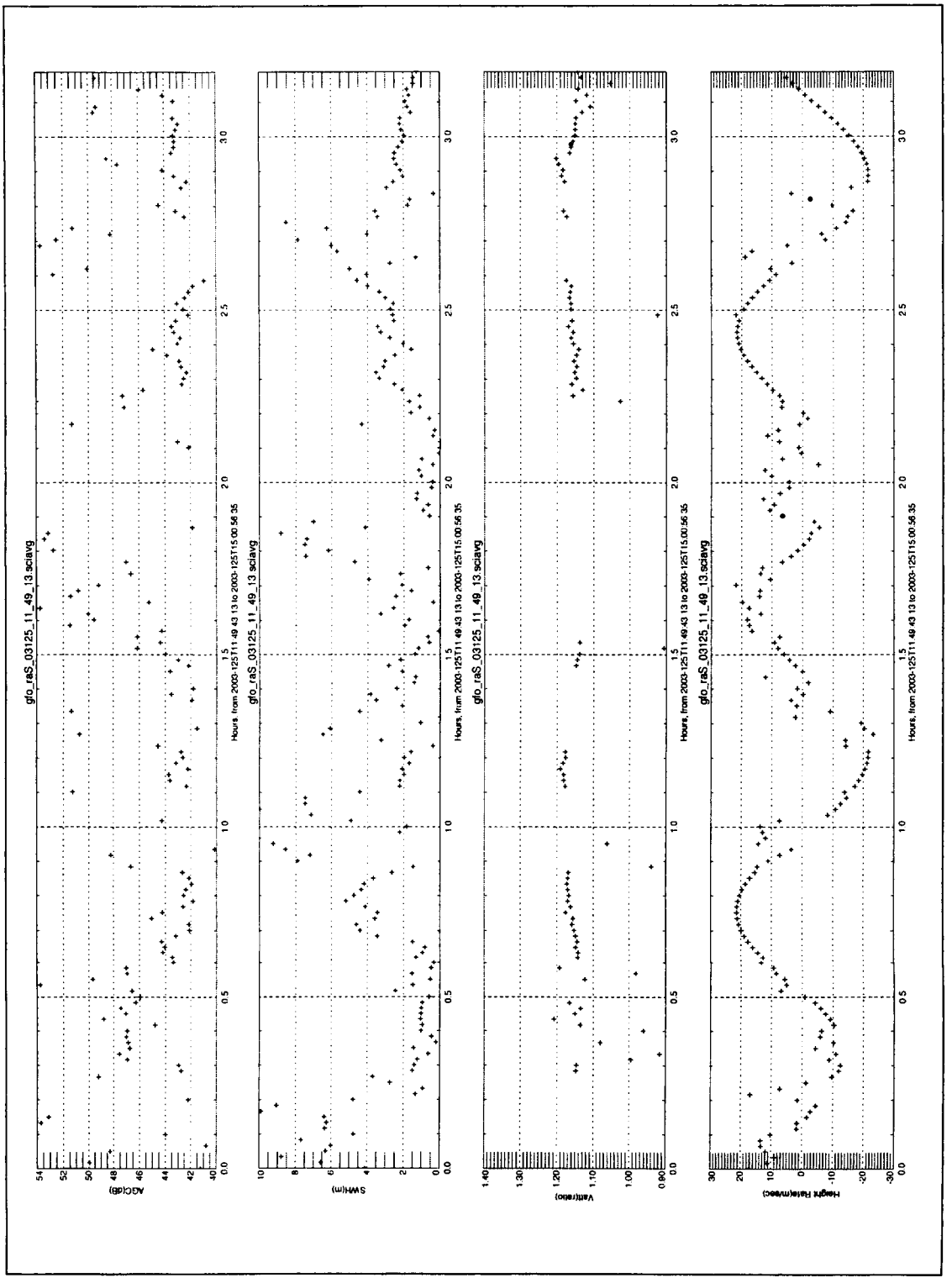

Figure B-2 RA Science Average Plot (Continued)

B.3 RA Engineering Average Plot

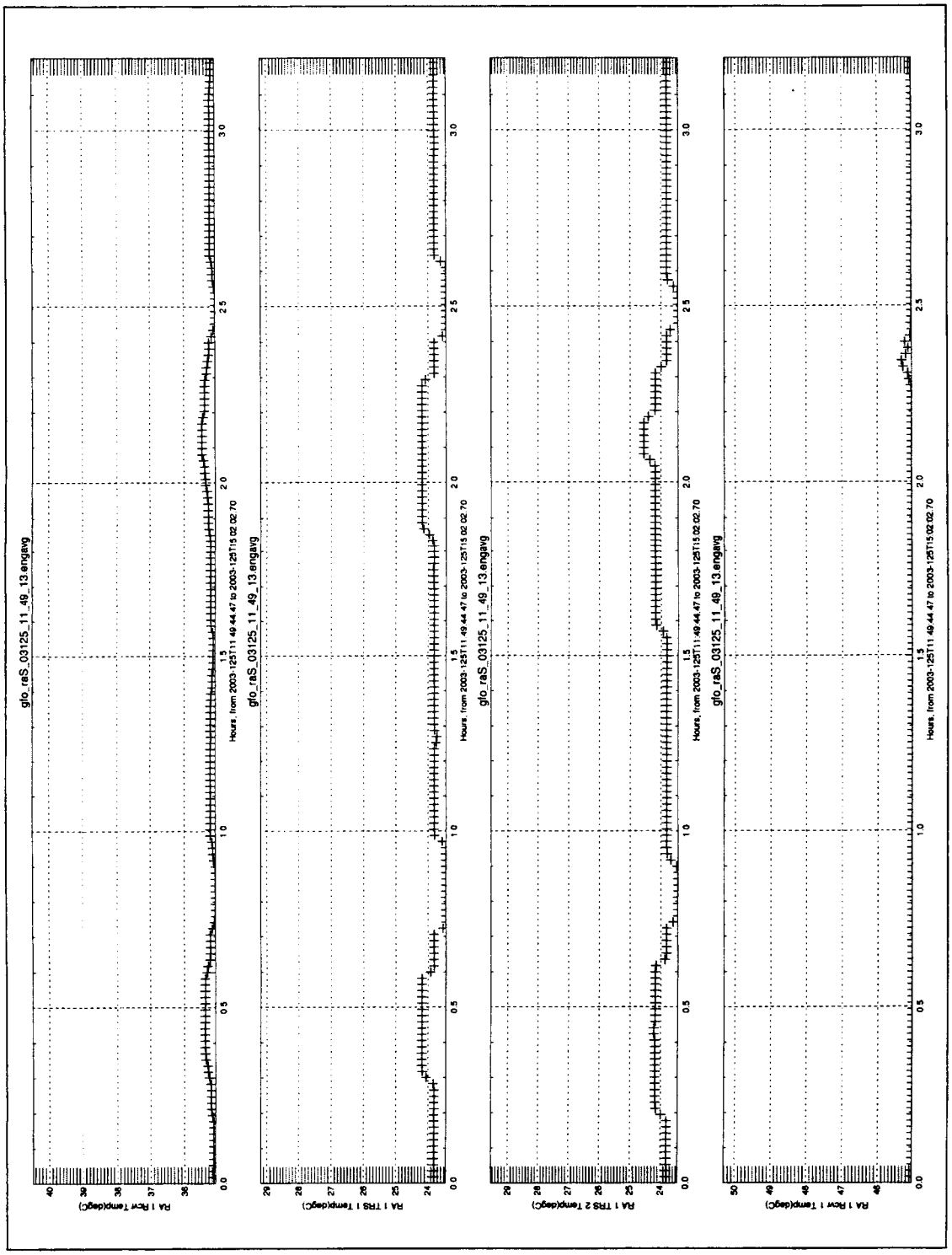

Figure B-3 RA Engineering Average Plot

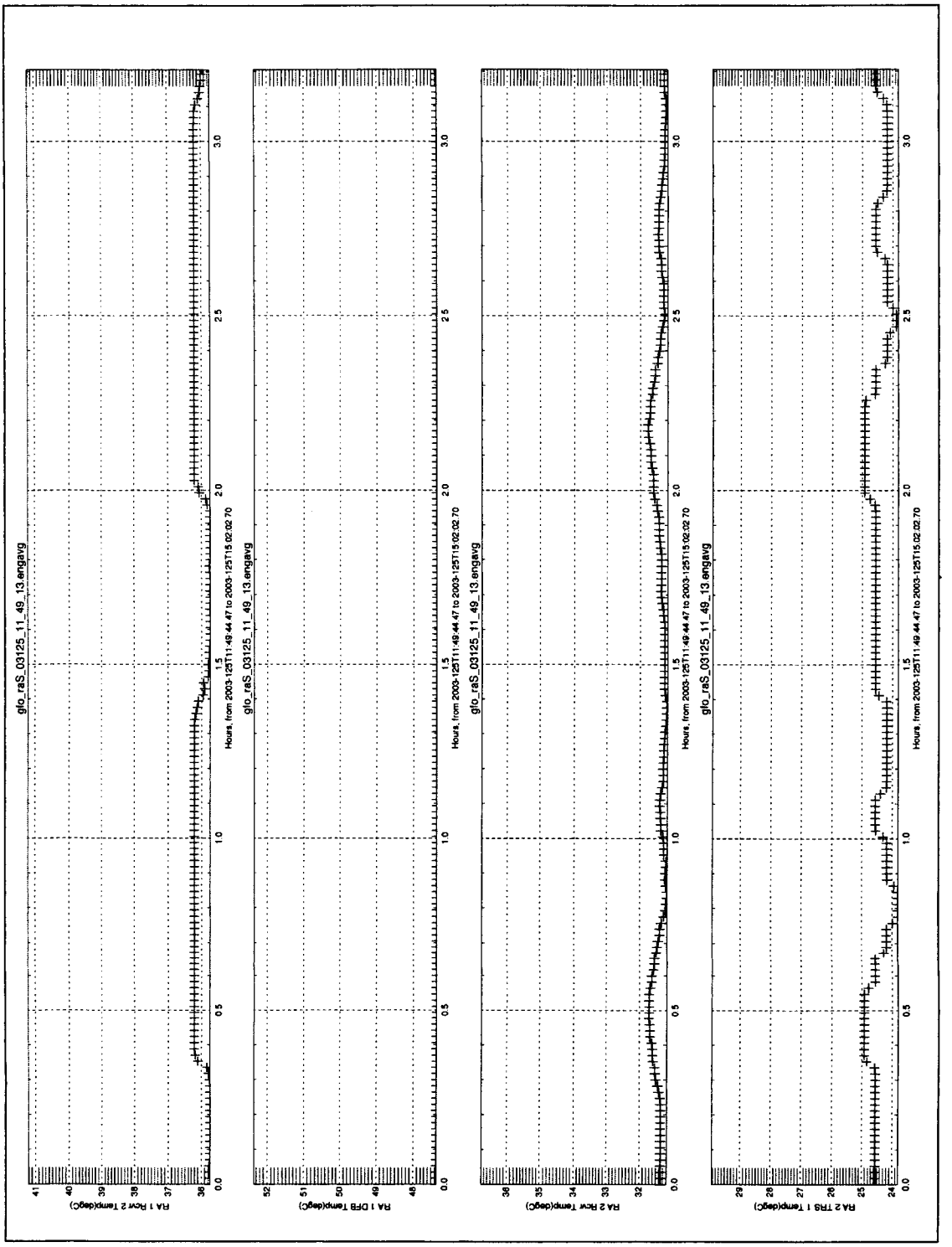

Figure B-3 RA Engineering Average Plot (Continued)

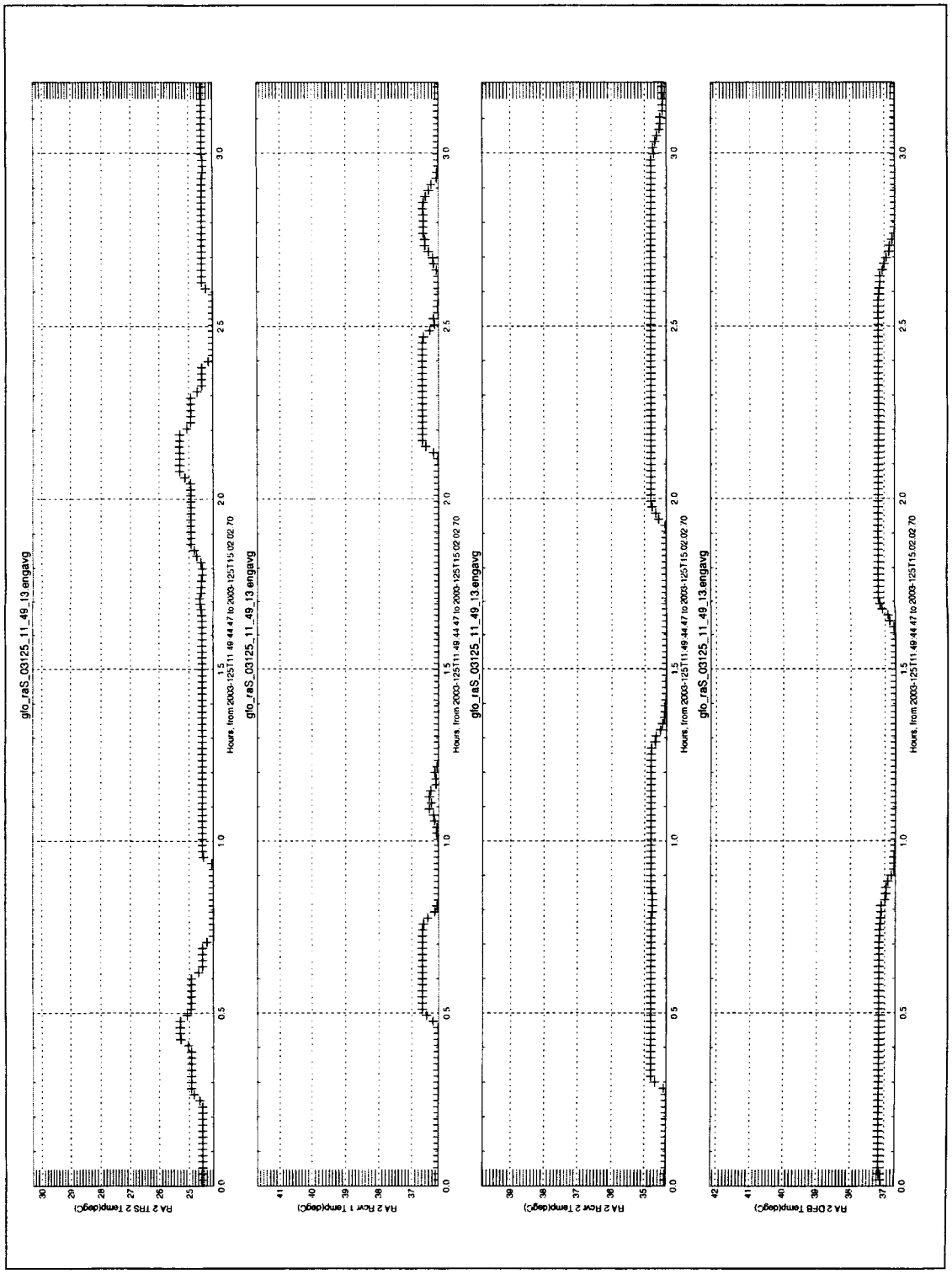

Figure B-3 RA Engineering Average Plot (Continued)

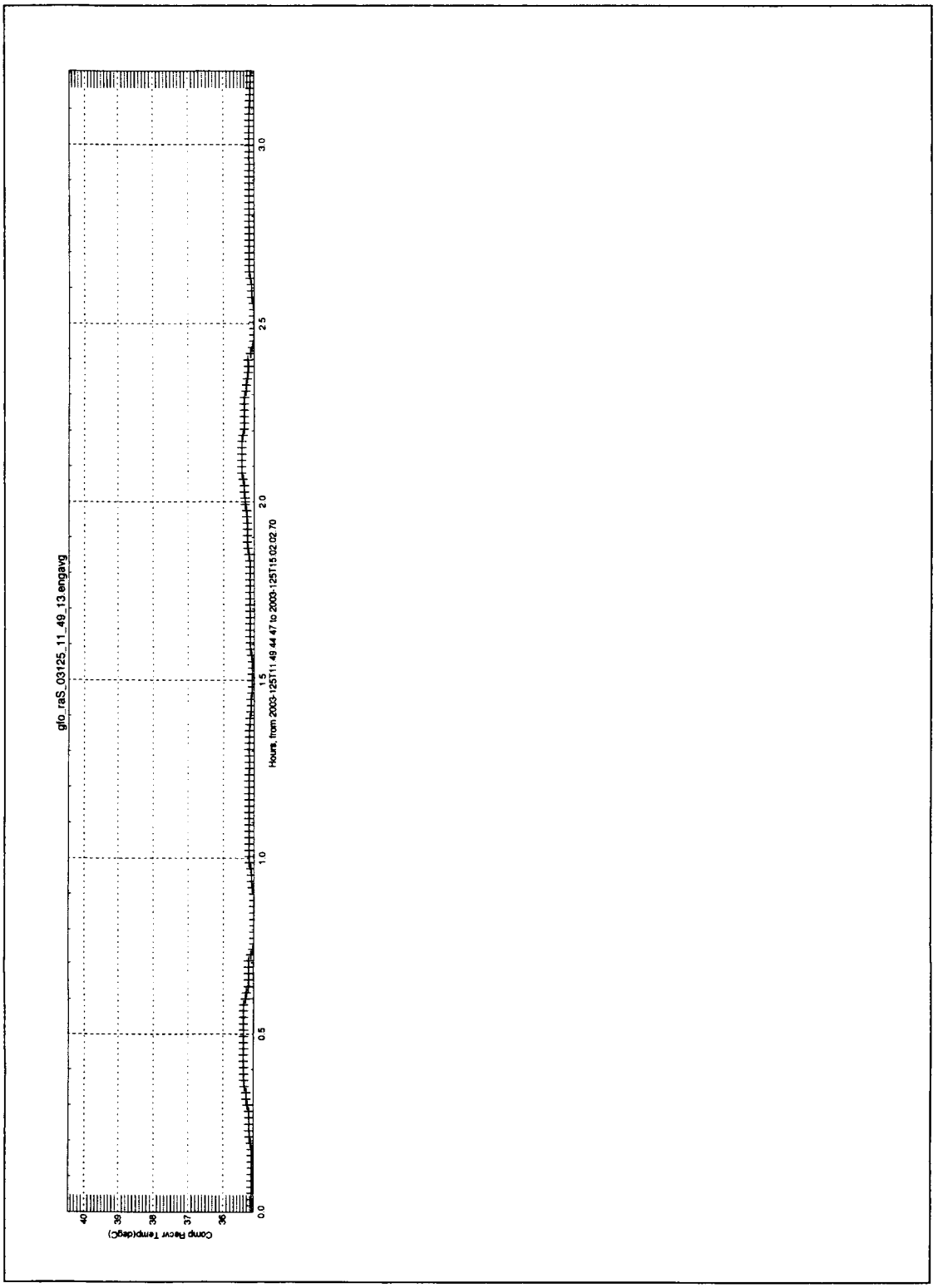

Figure B-3 RA Engineering Average Plot (Continued)

B.4 RA Event Listing

```
gfoEvent03125.dat

gfo_raL_03125_02_09_12.event:RA 1 SSPA  03125_02_0  105372552.4 2003-125T02:09:12 BIT/Test            FAILFirst
gfo_raL_03125_02_09_12.event:RA 1 SSPA  03125_02_0  105372552.4 2003-125T02:09:12 S/A Mode          ACTIVEFirst
gfo_raL_03125_05_31_40.event:RA 1 SSPA  03125_05_3  105384701.0 2003-125T05:31:40 BIT/Test            FAILFirst
gfo_raL_03125_05_31_40.event:RA 1 SSPA  03125_05_3  105384701.0 2003-125T05:31:40 S/A Mode          ACTIVEFirst
gfo_raS_03125_05_31_40.event:RA 1 SSPA  03125_05_3  105388046.7 2003-125T06:27:26 BIT/Test            FAILFirst
gfo_raS_03125_05_31_40.event:RA 1 SSPA  03125_05_3  105388046.7 2003-125T06:27:26 S/A Mode          ACTIVEFirst
gfo_raL_03125_05_31_40.event:RA 1 SSPA  03125_05_3  105388050.3 2003-125T06:27:30 TelemFmt   CAL1
gfo_raL_03125_05_31_40.event:RA 1 SSPA  03125_05_3  105388060.1 2003-125T06:27:40 TelemFmt   CAL2
gfo_raL_03125_05_31_40.event:RA 1 SSPA  03125_05_3  105388070.2 2003-125T06:27:50 CAL Constr Freeze
gfo_raL_03125_05_31_40.event:RA 1 SSPA  03125_05_3  105388662.0 2003-125T06:37:42 CAL Constr Normal
gfo_raS_03125_05_31_40.event:RA 1 SSPA  03125_05_3  105395850.1 2003-125T08:37:30 BIT/Test   FAIL
gfo_raS_03125_05_31_40.event:RA 1 SSPA  03125_05_3  105395850.1 2003-125T08:37:30 BIT/Test   NORM
gfo_raS_03125_11_49_13.event:RA 1 SSPA  03125_11_4  105407353.2 2003-125T11:49:13 BIT/Test            FAILFirst
gfo_raS_03125_11_49_13.event:RA 1 SSPA  03125_11_4  105407353.2 2003-125T11:49:13 S/A Mode          ACTIVEFirst
gfo_raS_03125_15_02_20.event:RA 1 SSPA  03125_15_0  105418940.7 2003-125T15:02:20 BIT/Test            FAILFirst
gfo_raS_03125_15_02_20.event:RA 1 SSPA  03125_15_0  105418940.7 2003-125T15:02:20 S/A Mode          ACTIVEFirst
gfo_raS_03125_21_43_02.event:RA 1 SSPA  03125_21_4  105442982.1 2003-125T21:43:02 BIT/Test            FAILFirst
gfo_raS_03125_21_43_02.event:RA 1 SSPA  03125_21_4  105442982.1 2003-125T21:43:02 S/A Mode          ACTIVEFirst
```

Figure B-4 RA Event Listing

B.5 SDR Log Listing

```
FileName: sdr03125_02_09_13_12395.dat
      ***Date Data Used***
    StartYear: 2003
     StartDOY: 125
    StartHour:  2
     StartMin:  9
     StartSec: 13
     StartUTC:   7753.497239655
      StopUTC:  19899.623235033
      ***Date Data Measured***
      UTCYear: 2003
       UTCDOY: 125
   UTCSeconds:  13274.006662000
    VTCWValue:  94900139999136.00
        Ratio:   0.0000009999198161375794

---------------------------------


      Num RecordZeroFilled:      0 D
             Num NotInFTRK:   4348 D
        Num BackScatterErr:    518
           Num RcvrTempErr:      0
            Num VATTEstErr:   6013
        Num NoSmoothedVATT:   5150 D
                Num RateErr:    510
          Num SWHBoundsErr:      0 D
          Num AGCBoundsErr:    110
          Num HgtBoundsErr:    110
            Num DFBTempErr:      0
          Num Rcvr2TempErr:      0
          Num Rcvr1TempErr:      0
           Num Trs2TempErr:      0
           Num Trs1TempErr:      0
           Num OffNadirErr:      0 D
             Num SWHSTDErr:   1885 D
             Num AGCSTDErr:   3412
          Num HeightSTDErr:   5397
     Num FiveFramesMissing:      5 D

      Total Frames Deleted:   5342

         Total Frames Used:   7053

              Total Frames:  12395

          Total BS0Delete:   5151
        Total BS0NotDelete:     30
```

Figure B-5 SDR Log Listing

B.6 SDR Science Average Plot

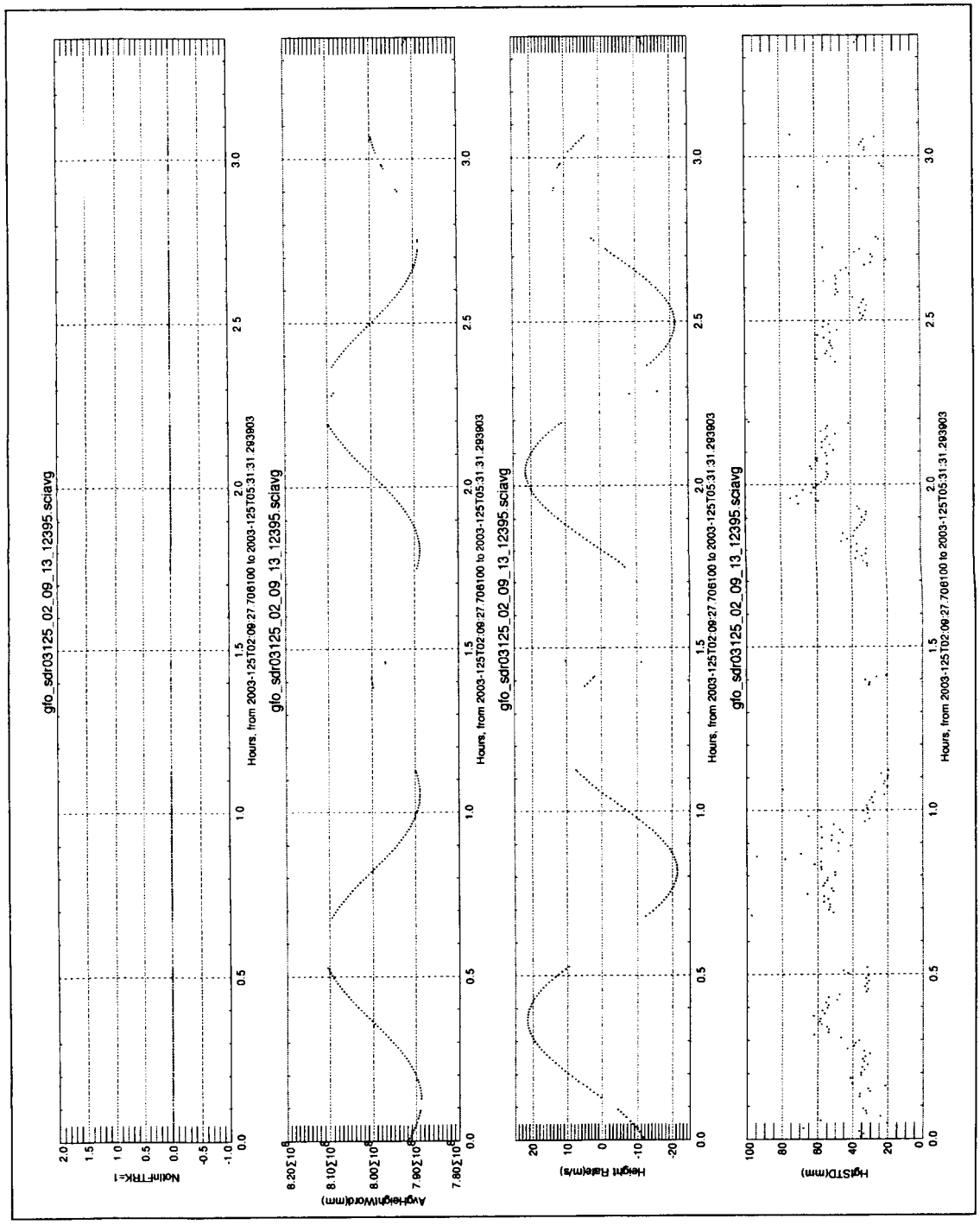

Figure B-6 SDR Science Average Plot

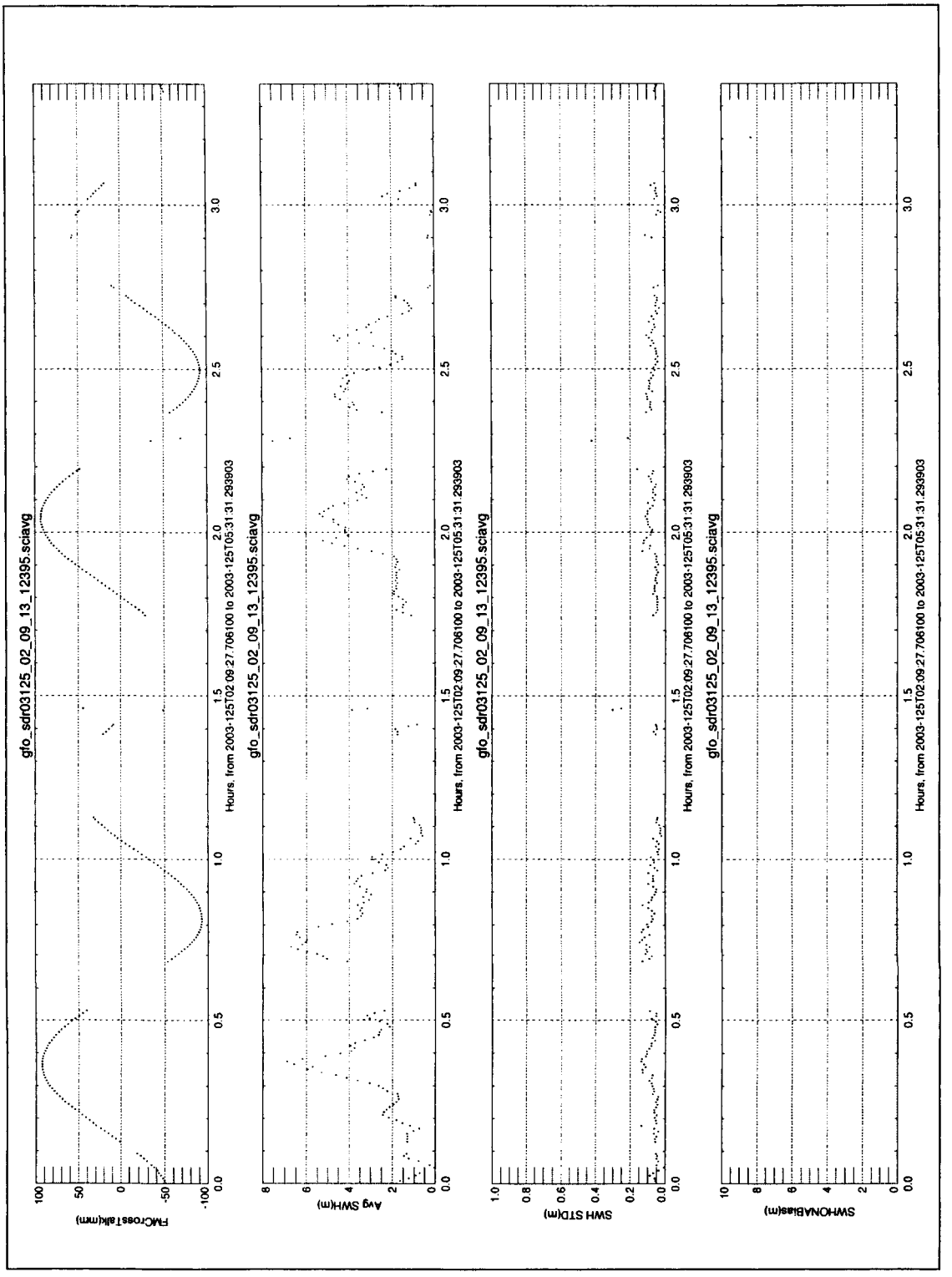

Figure B-6 SDR Science Average Plot (Continued)

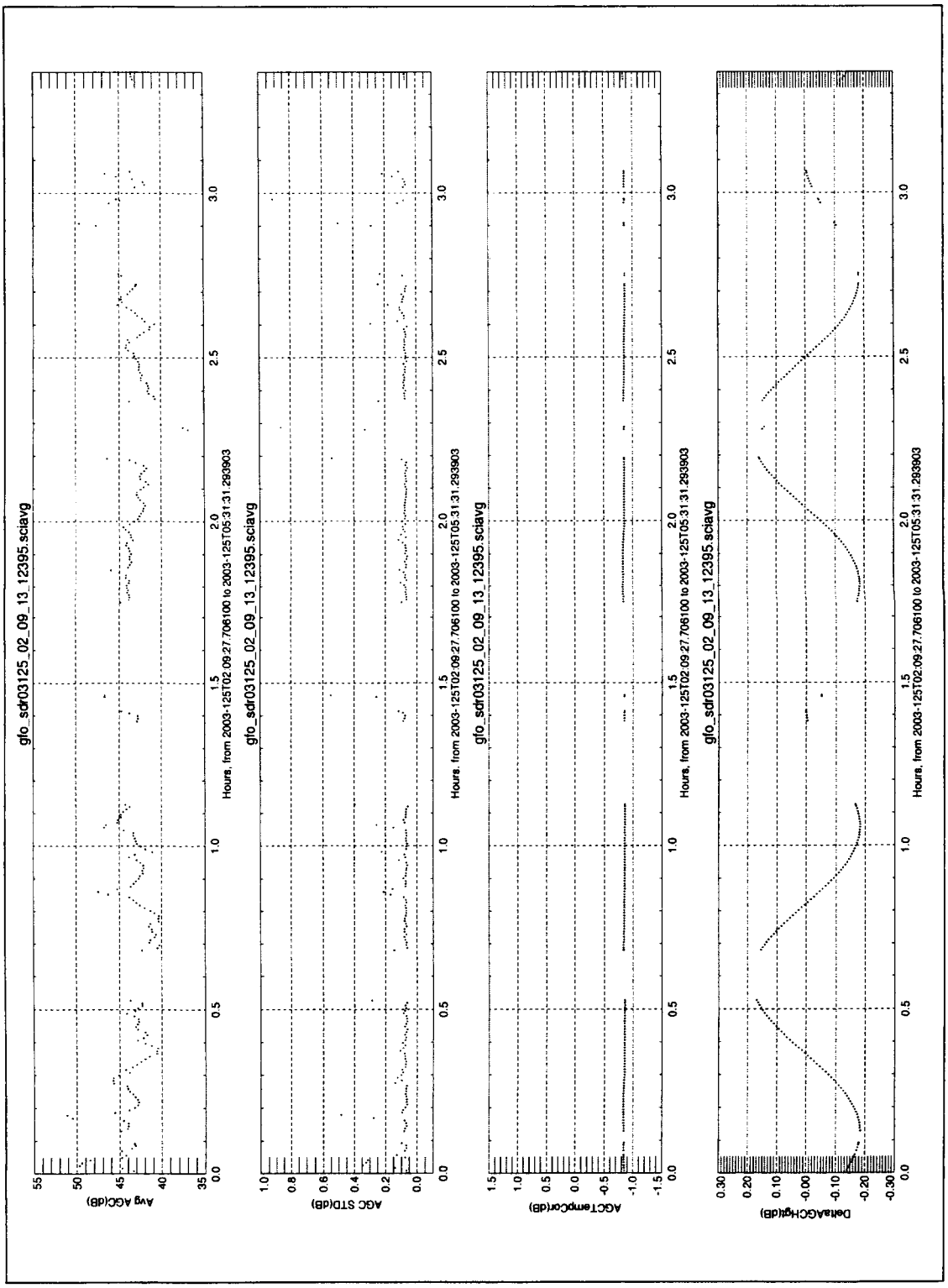

Figure B-6 SDR Science Average Plot (Continued)

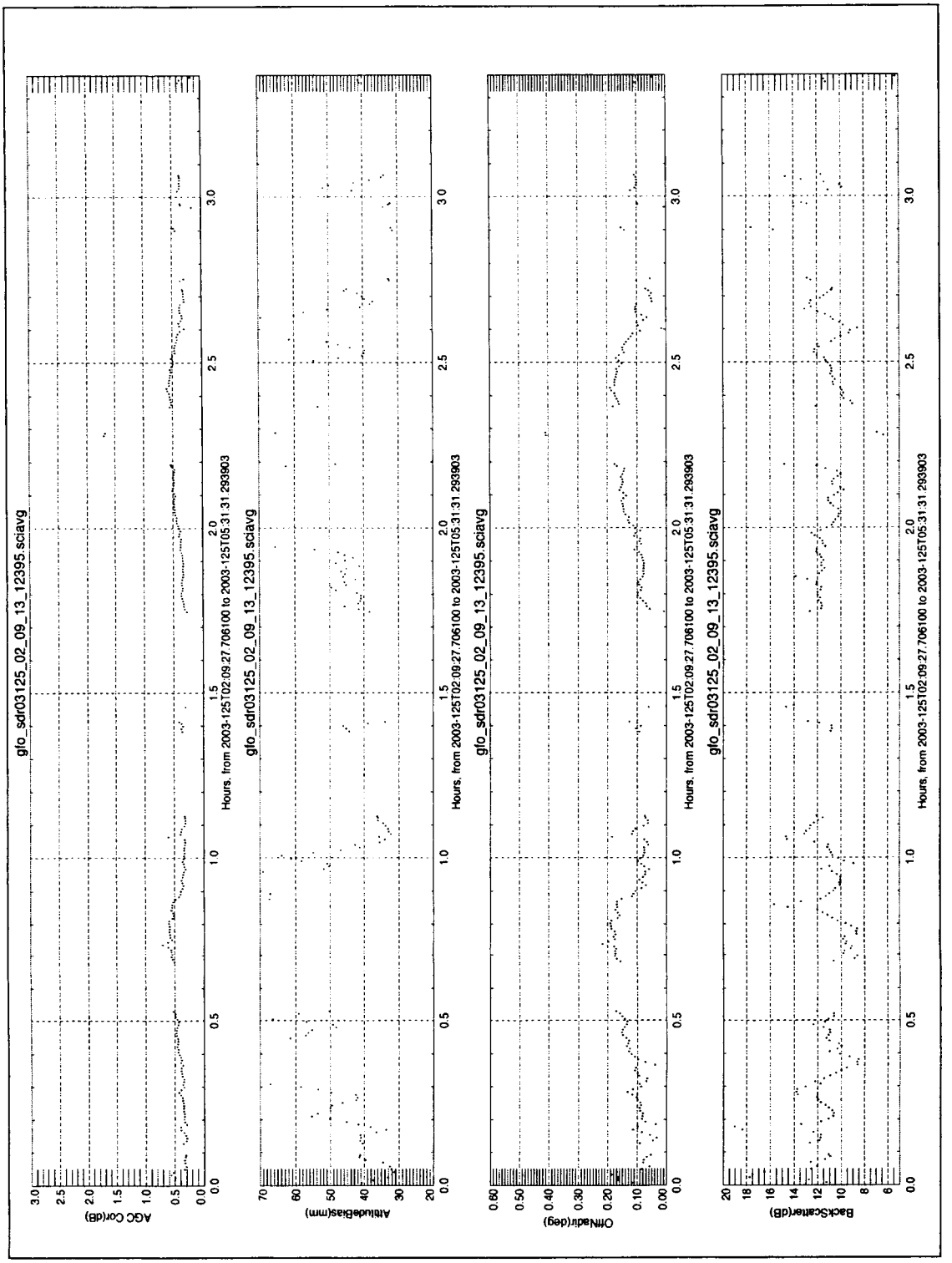

Figure B-6 SDR Science Average Plot (Continued)

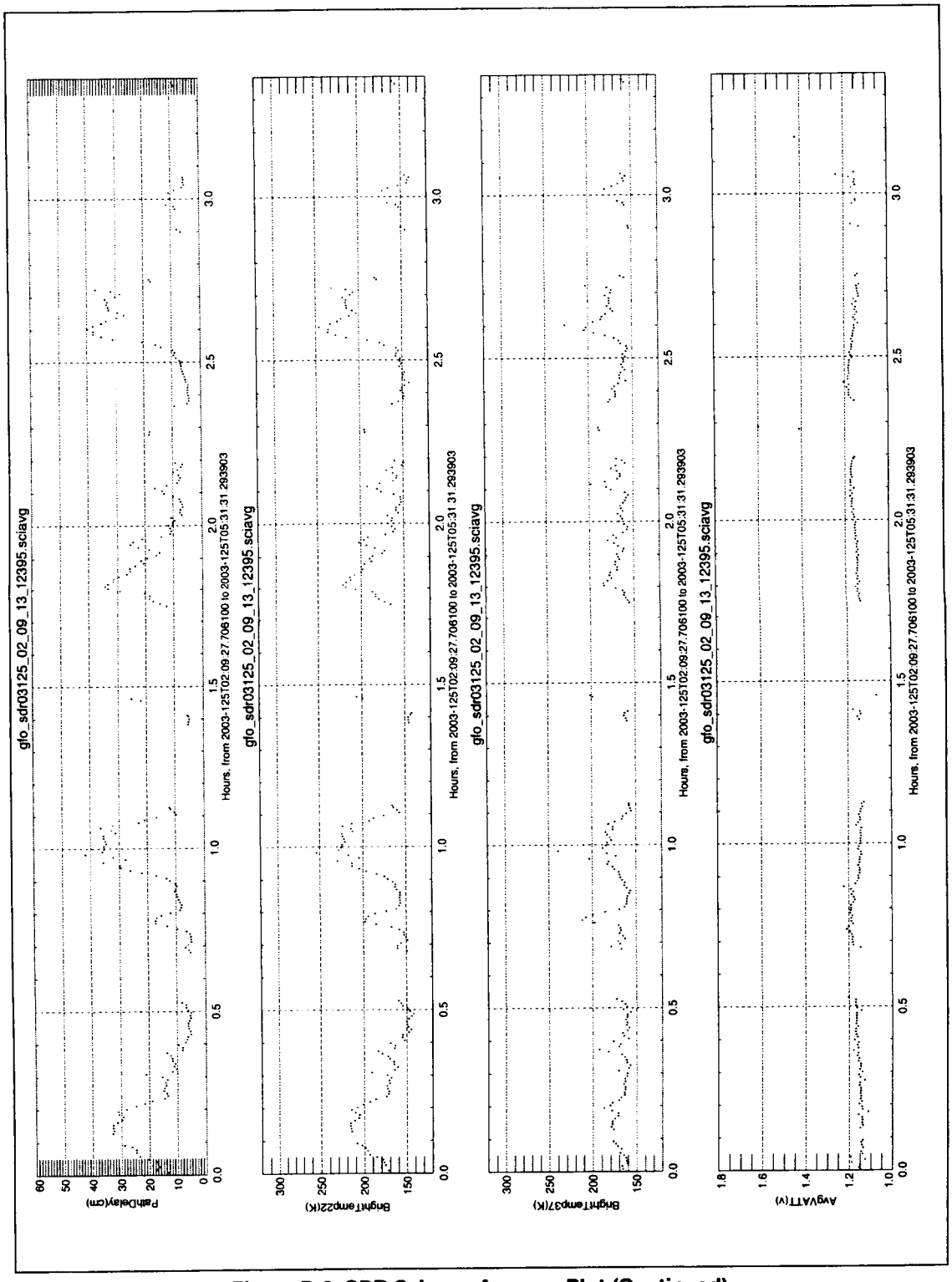

Figure B-6 SDR Science Average Plot (Continued)

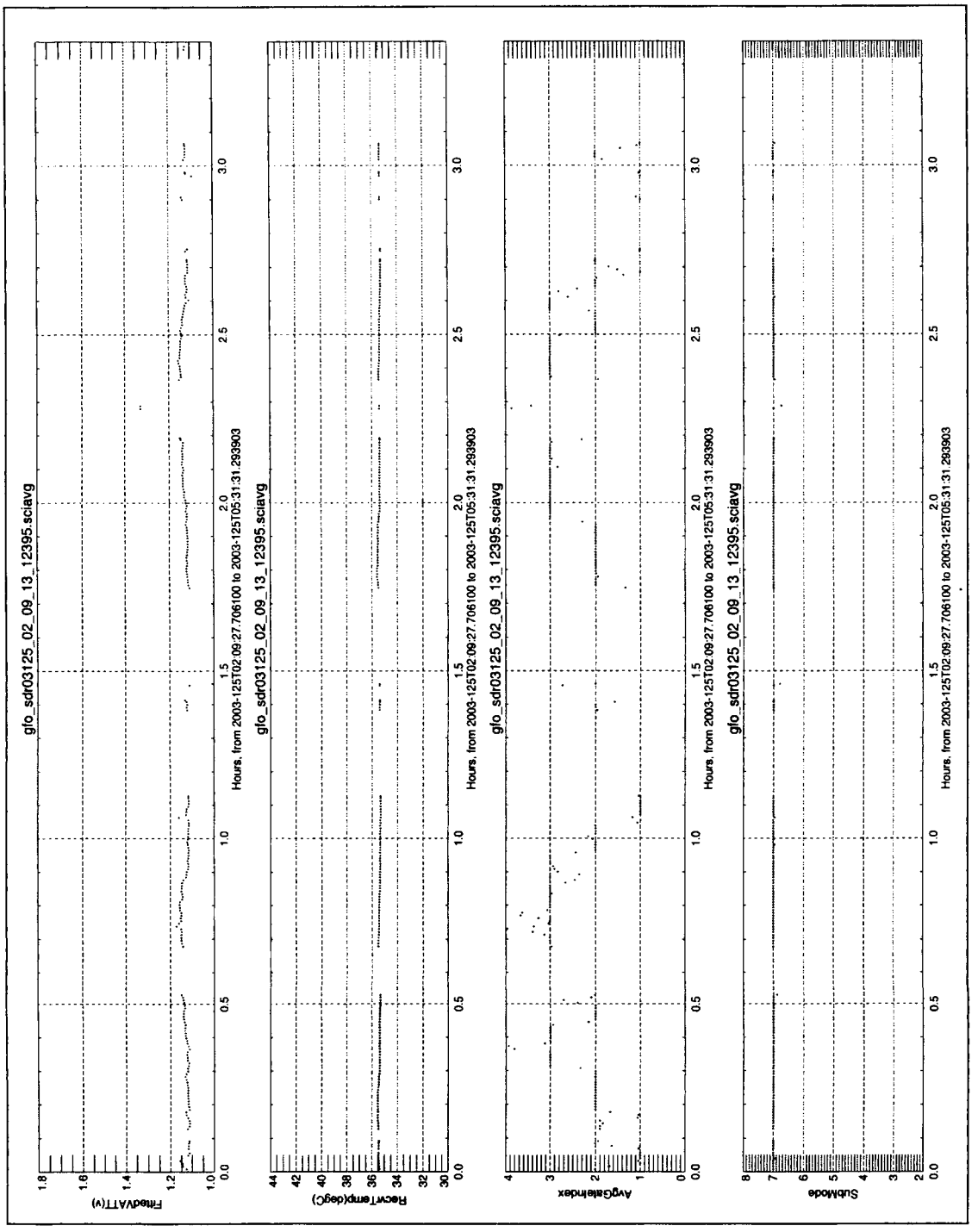

Figure B-6 SDR Science Average Plot (Continued)

B.7 SDR Quicklook Plot

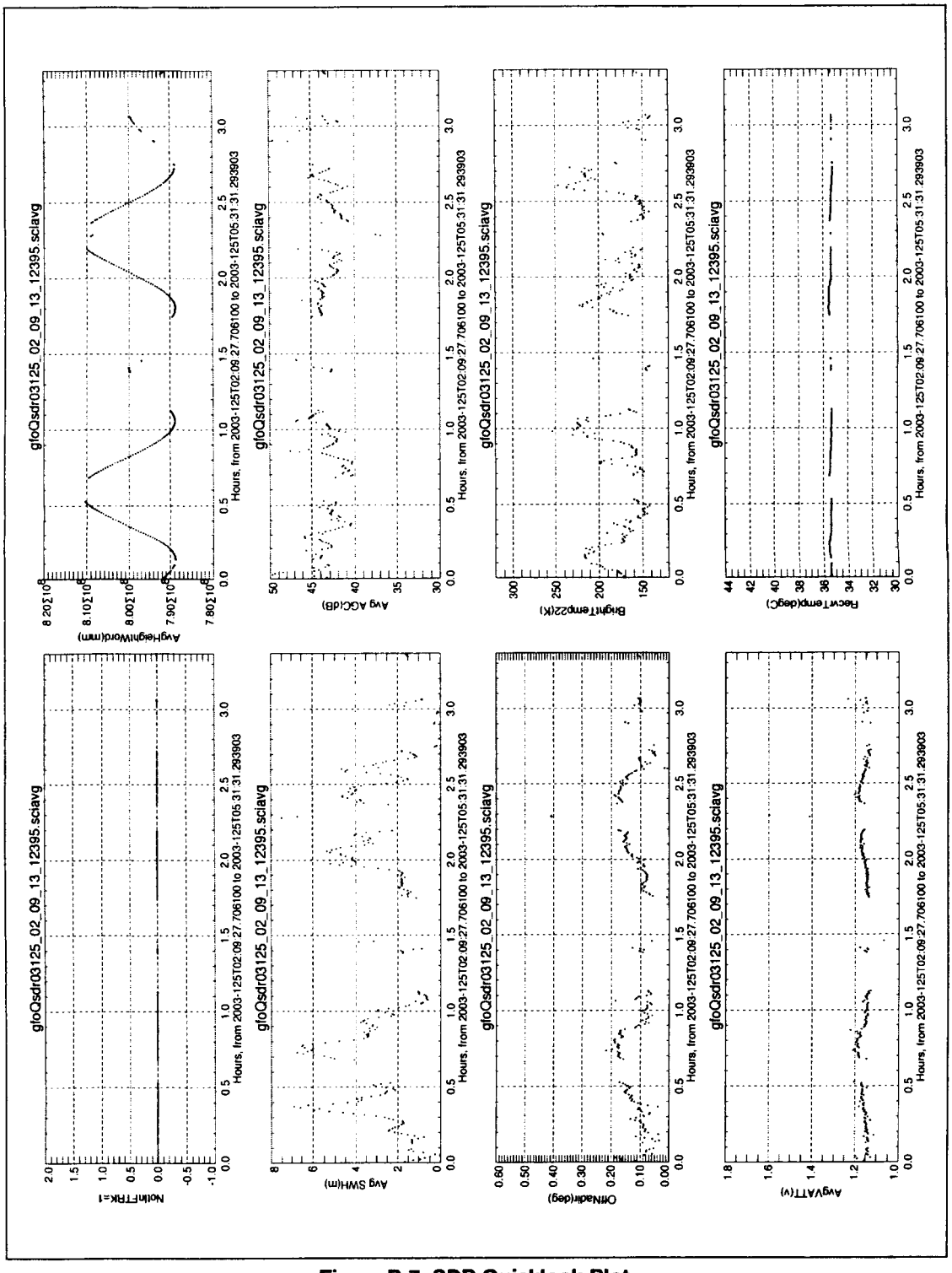

Figure B-7 SDR Quicklook Plot

B.8 NGDR Log Listing

```
--------------------------------------------------------------------------
                          NGDR File Header
--------------------------------------------------------------------------

                   PASS_BEGIN_TIME =     0.35896442;
                  REVOLUTION_NUMBER = 2147483647;
                    CYCLE_NUMBER = 2147483647;
                     PASS_NUMBER = 2147483647;
   PROCESSING_TIME =  6699.32834491 2003-126T07:52:49 (2003-05-06) UTC;
                   PROCESSING_CENTER = NAVO ADFC;
              SOFTWARE_VERSION = Software Version = 1.0;
                         SATELLITE_ID = GFO;
                    DATA_RECORD_LENGTH = 184;
                     BASIC_GDR_LENGTH = 98;
                HEIGHT_CALIBRATION_BIAS = 0.000000;
                 ALTITUDE_BIAS_INITIAL = 0.020815;
            ALTITUDE_BIAS_CENTER_OF_GRAVITY = 292.000000;
                   SWH_BIAS_INITIAL = 0.000000;
                 AGC_CALIBRATION_BIAS = -0.370000;
                   AGC_BIAS_INITIAL = 31.860001;
              ORB File = oodd03123_23_19_00_3961.dat;

--------------------------------------------------------------------------
Delta SciTime Gap   105370649.999880   105371263.430714      613.43083405495
  5970
Delta SciTime Gap   105372551.047441   105372553.987205      2.9397640228271
  7285
Delta SciTime Gap   105384699.133279   105384702.073044      2.9397649765015
  19680
Delta SciTime Gap   105388050.464534   105388662.915412      612.45087790489
  23098
Delta SciTime Gap   105407351.976759   105407353.936602      1.9598430395126
  42171
Delta SciTime Gap   105418938.567609   105418941.507373      2.9397640228271
  53994
Delta SciTime Gap   105442980.939643   105442982.899486      1.9598429203033
  78527

       Num Frames in Track:  57864
            Hours in Track:  15.73
```

Figure B-8 NGDR Log Listing

```
------------------------------------------------------------------------
         Num LandContaminate:  13232 D       1162
        Num RecordZeroFilled:      2 D          2
              Num NotInFTRK:  29049 D          0


Num BackScatterErr:   3415         3225
             Num RcvrTempErr:      2 D          2
              Num VATTEstErr:  40746         11850
           Num NoSmoothedVATT:  34743 D       6069
                  Num RateErr:   2950          2085
            Num SWHBoundsErr:      0 D          0
            Num AGCBoundsErr:    688           18
            Num HgtBoundsErr:    688           18
              Num DFBTempErr:      0            0
            Num Rcvr2TempErr:      0            0
            Num Rcvr1TempErr:      0            0
             Num Trs2TempErr:      2            2
             Num Trs1TempErr:      0            0
             Num OffNadirErr:      0 D          0
               Num SWHSTDErr:  13070 D       4485
               Num AGCSTDErr:  22298          2316
            Num HeightSTDErr:  35726          6827
       Num FiveFramesMissing:     22 D          3

        Total Frames Deleted:      0            0

          Total Frames Used:  86913

                Total Frames:  86913
```

Figure B-8 NGDR Log Listing (Continued)

B.9 NGDR Day Plot

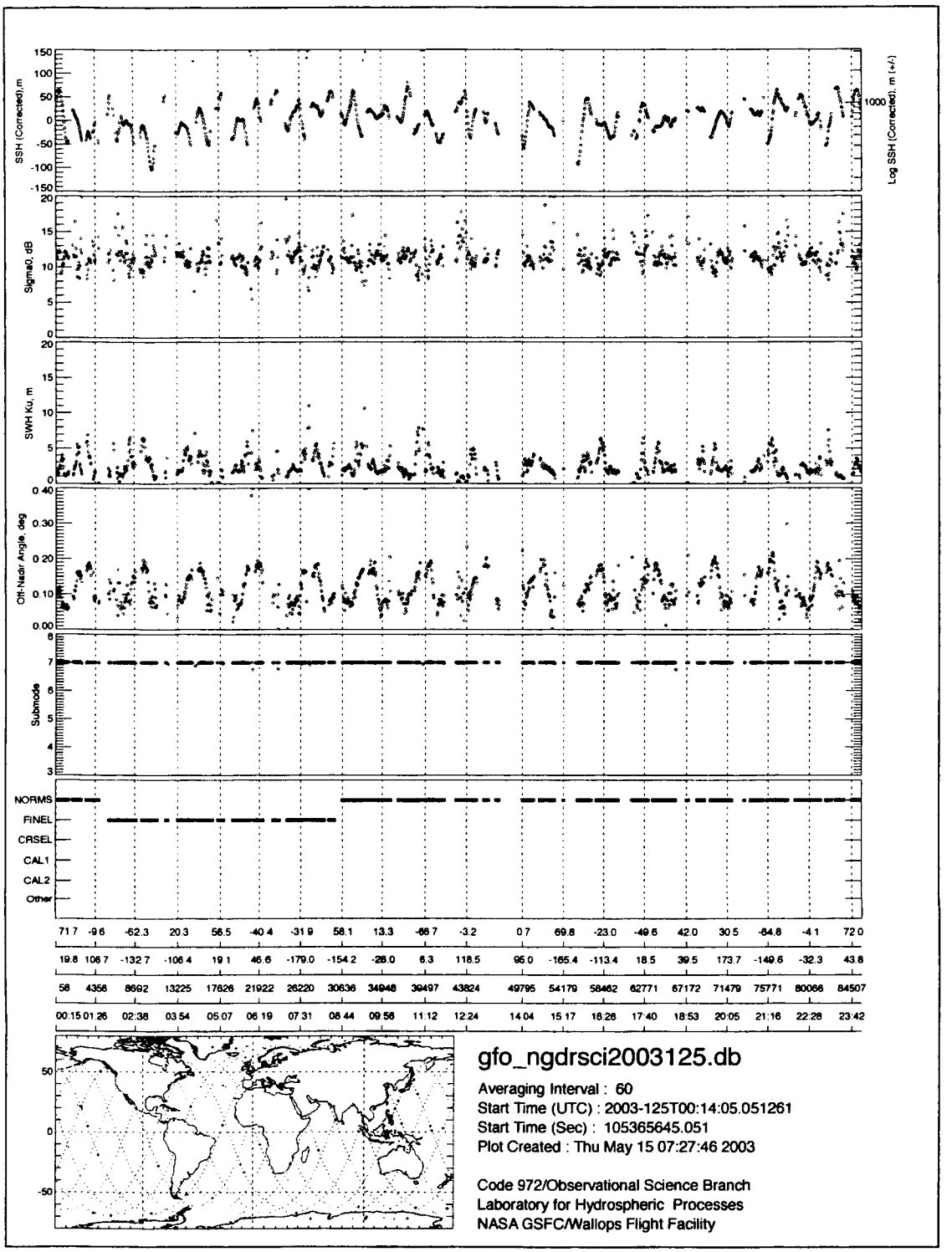

Figure B-9 NGDR Day Plot

B.10 NGDR Science Average Plot

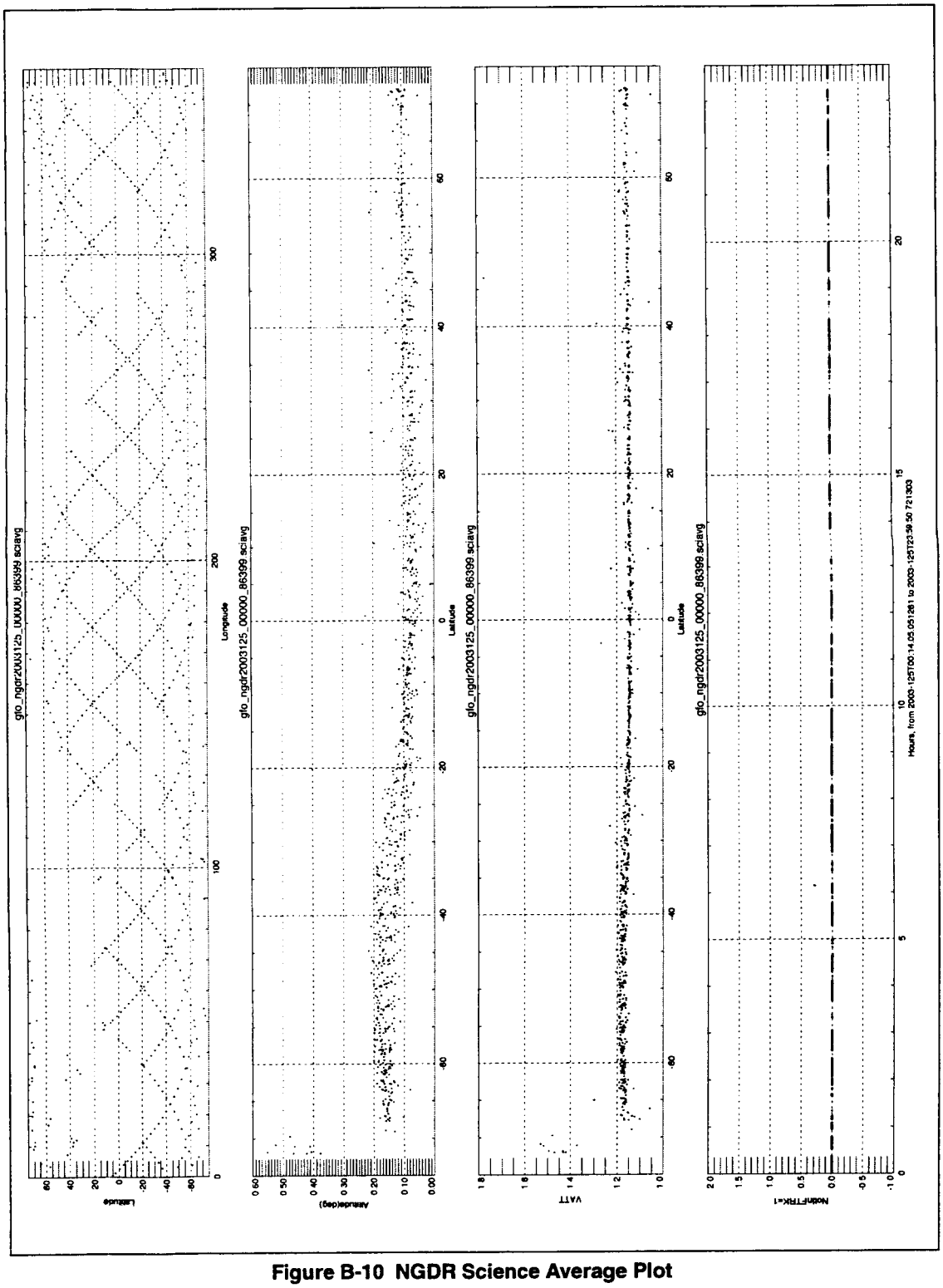

Figure B-10 NGDR Science Average Plot

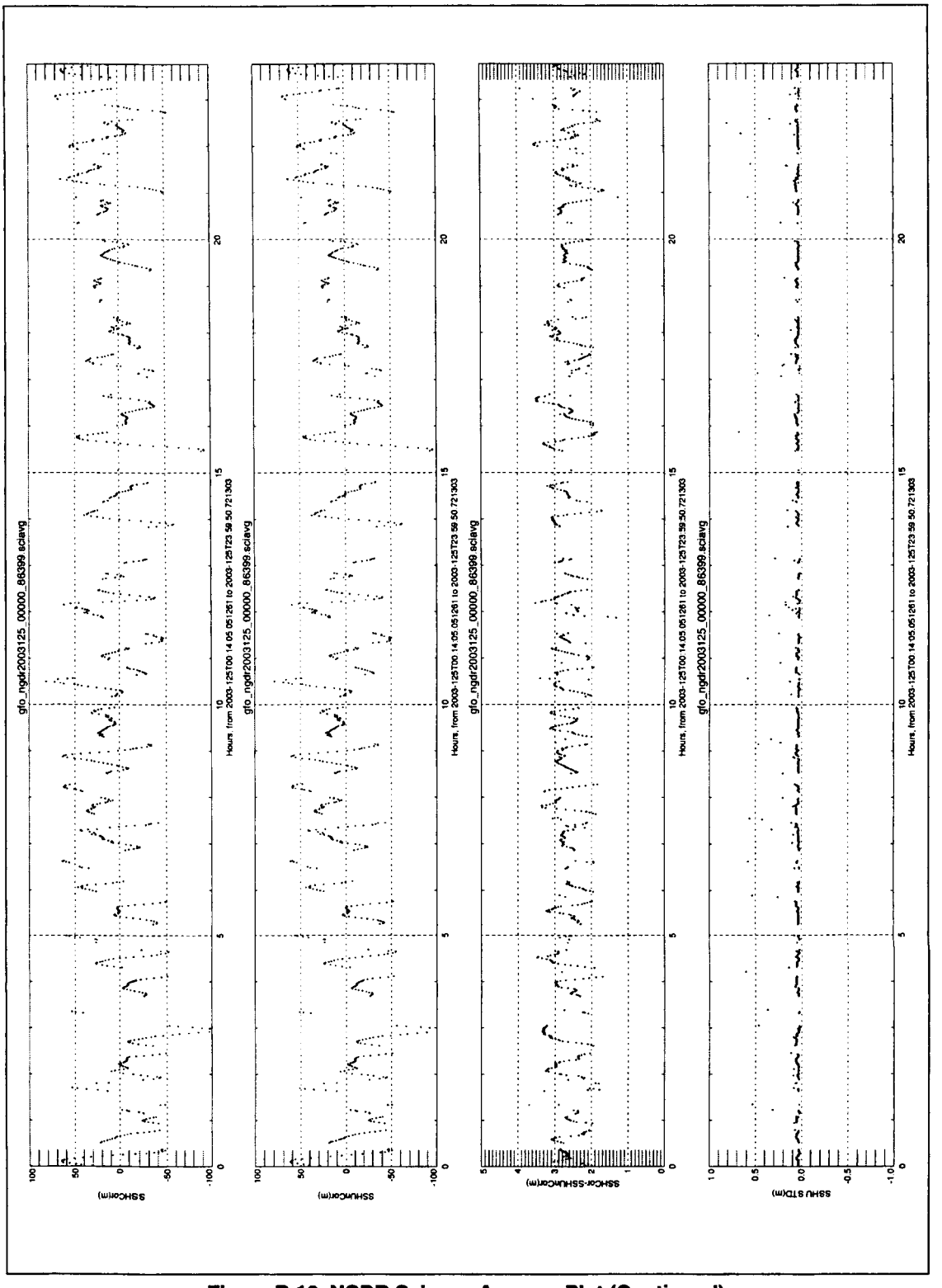

Figure B-10 NGDR Science Average Plot (Continued)

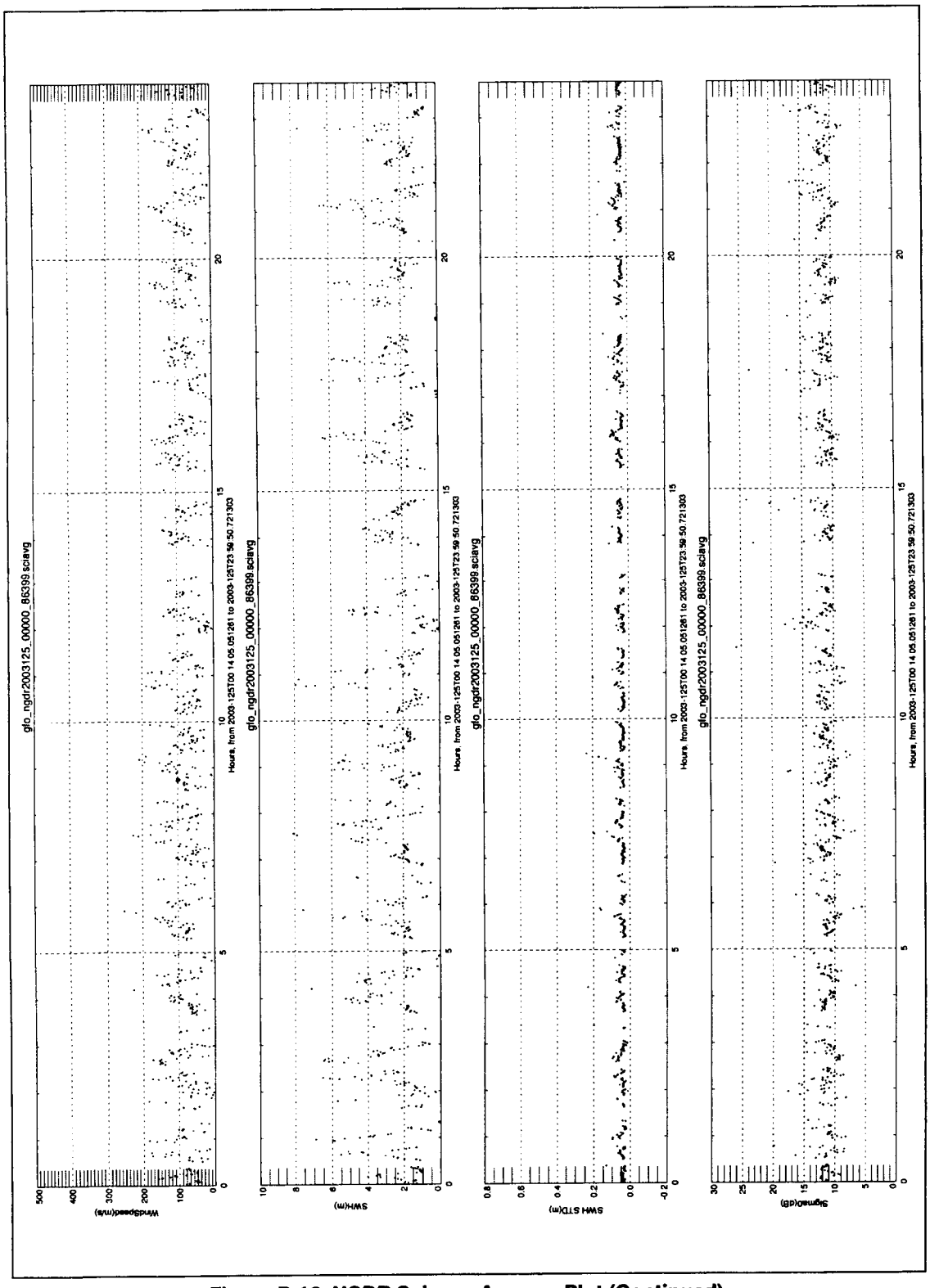

Figure B-10 NGDR Science Average Plot (Continued)

Figure B-10 NGDR Science Average Plot (Continued)

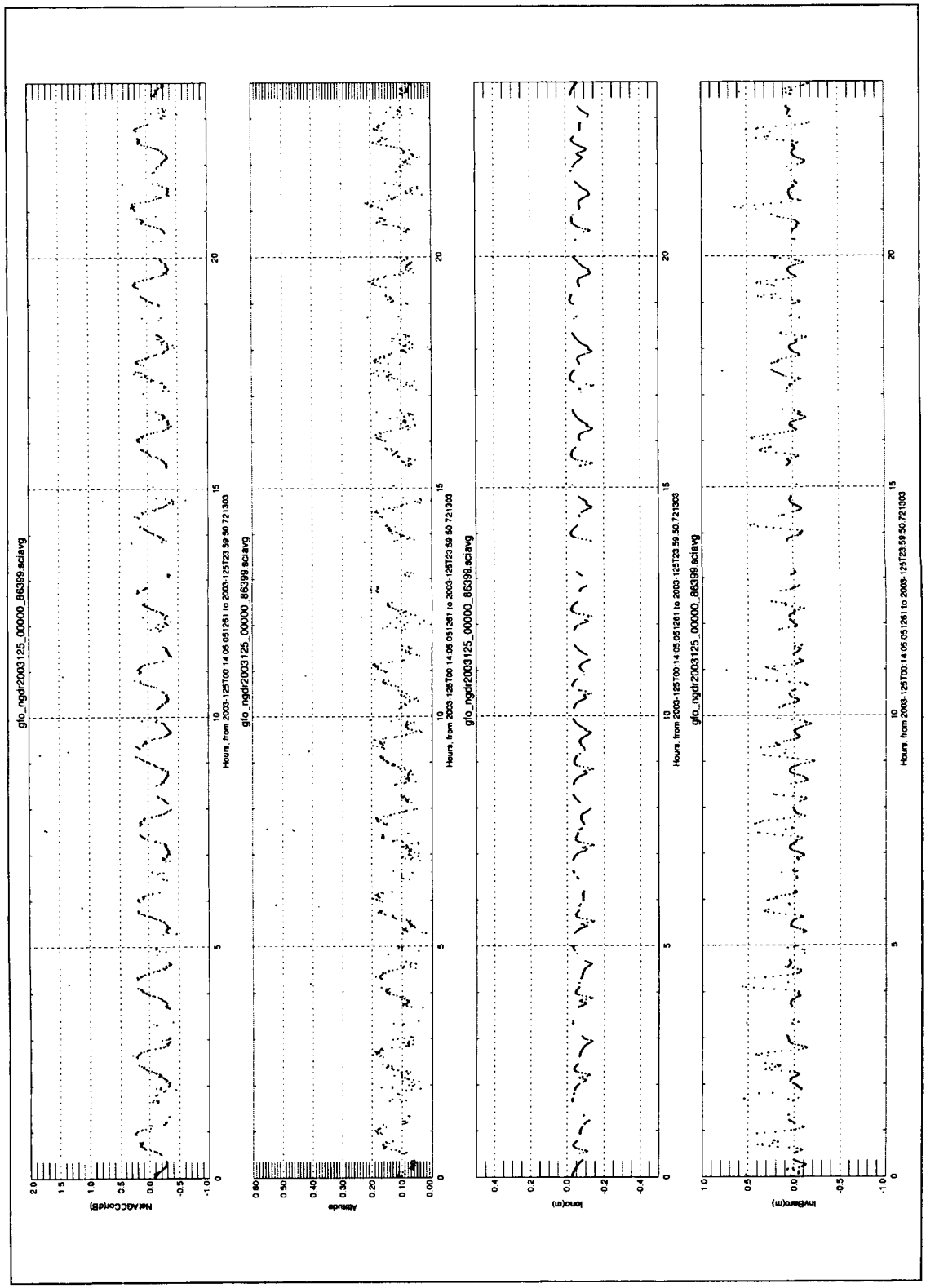

Figure B-10 NGDR Science Average Plot (Continued)

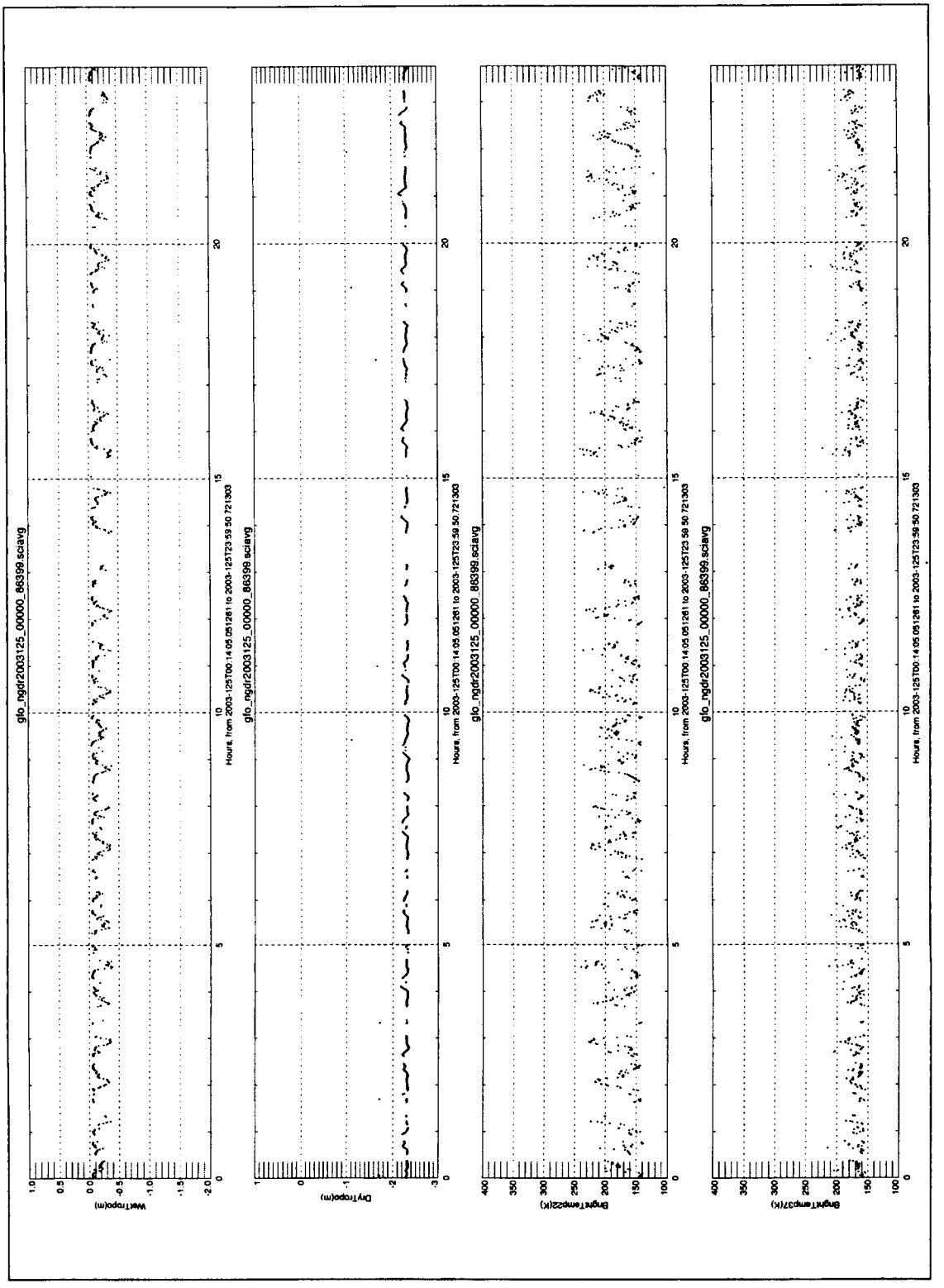

Figure B-10 NGDR Science Average Plot (Continued)

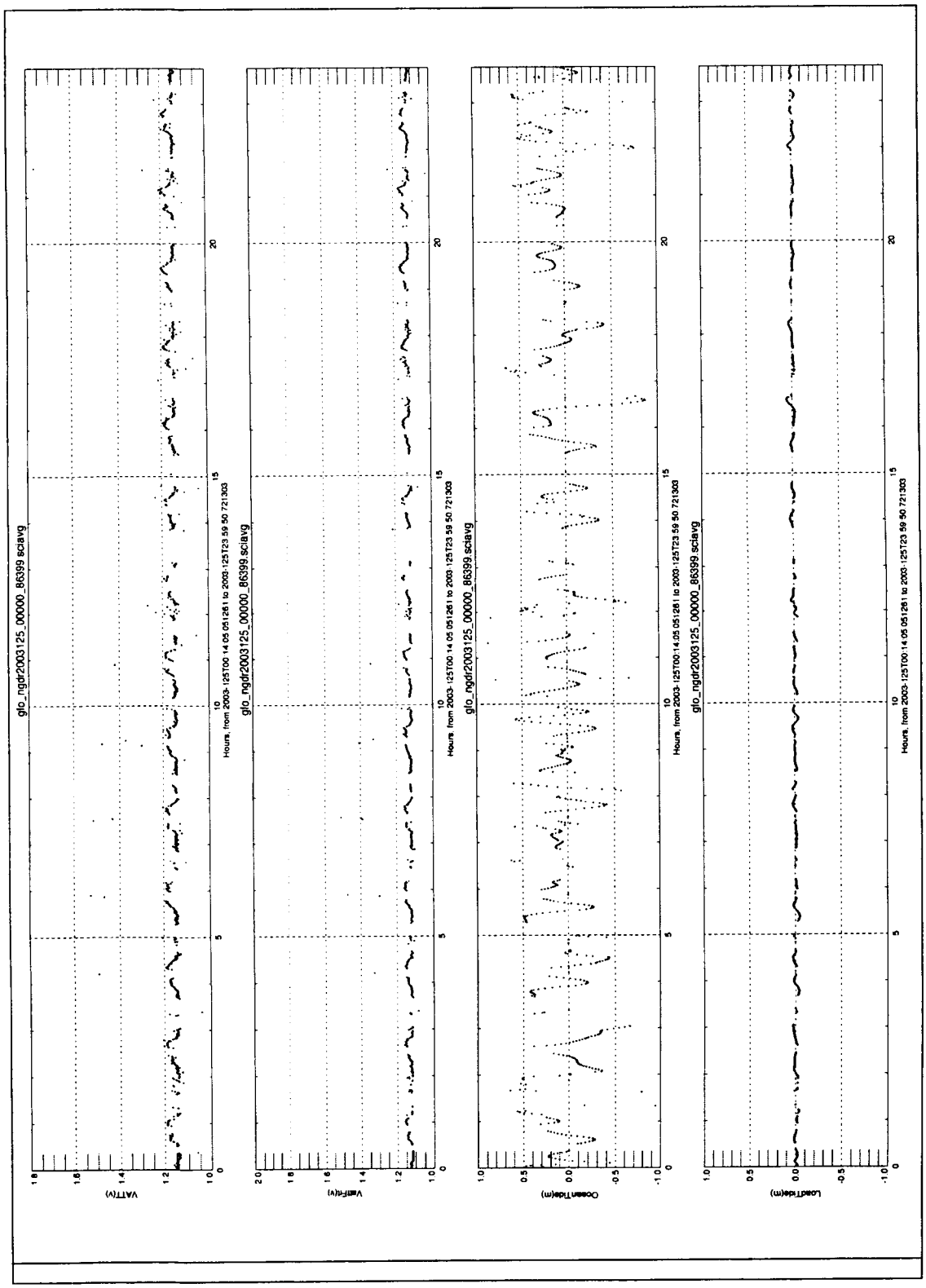

Figure B-10 NGDR Science Average Plot (Continued)

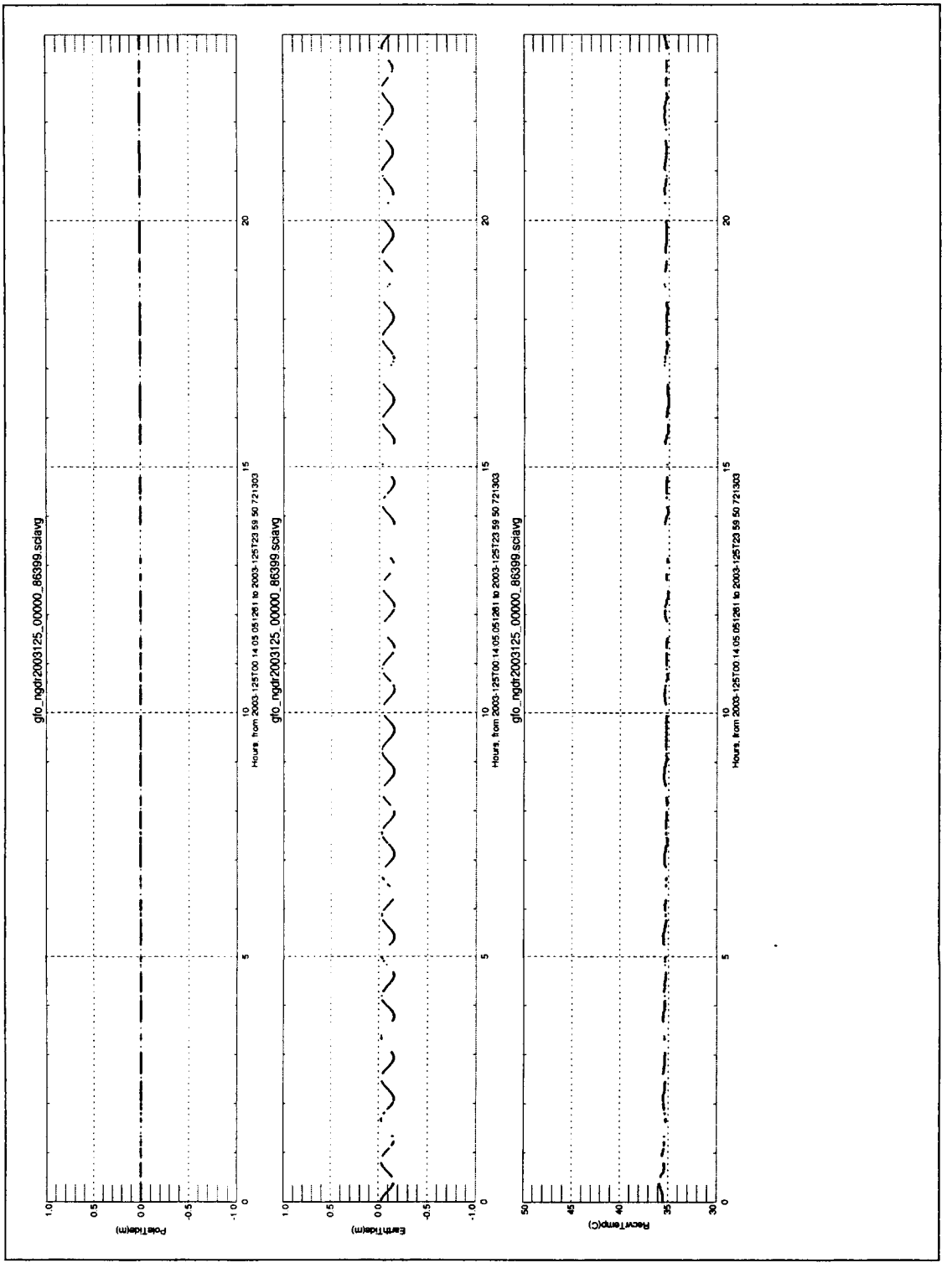

Figure B-10 NGDR Science Average Plot (Continued)

B.11 NGDR History Cycle Plot

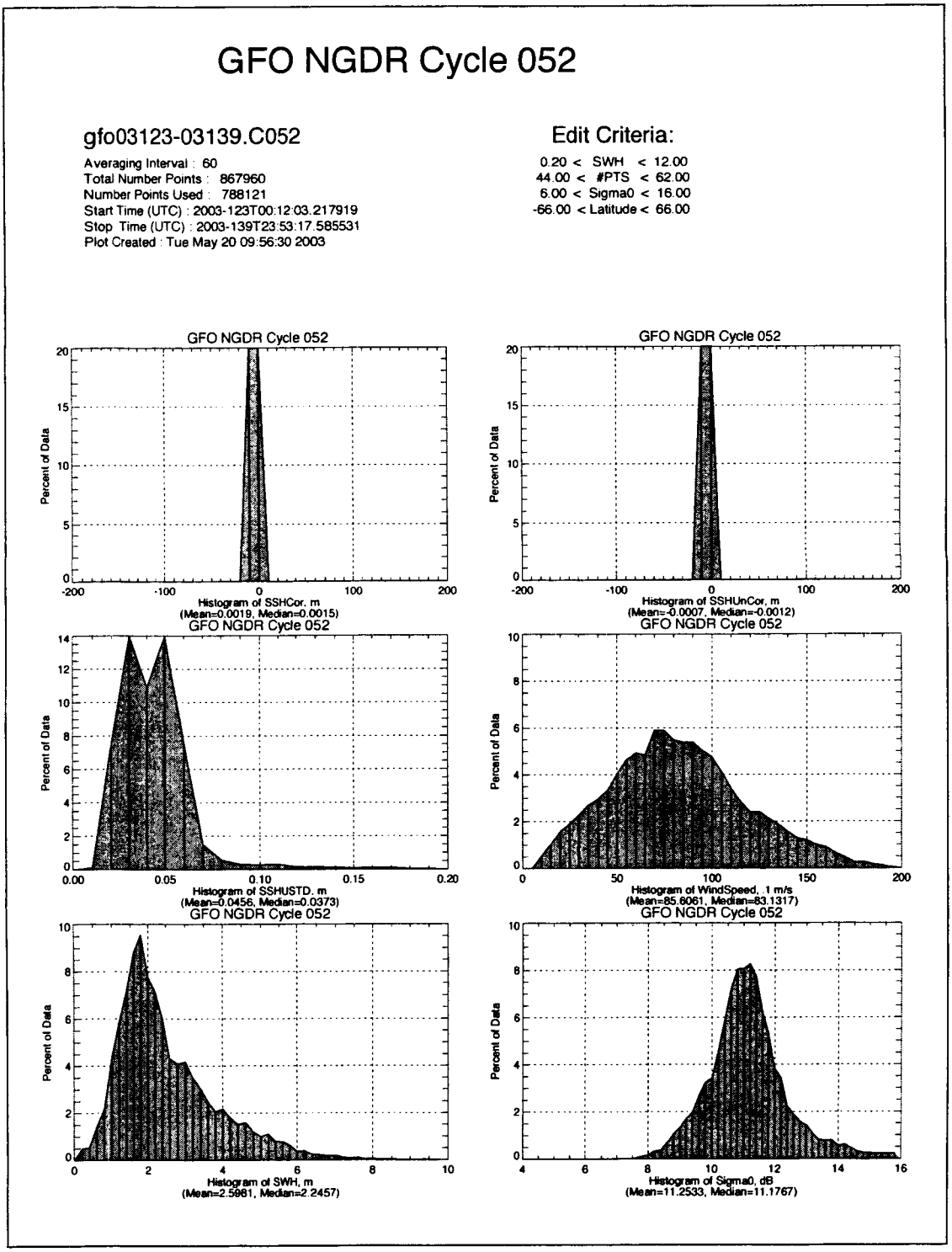

Figure B-11 NGDR History Cycle Plot

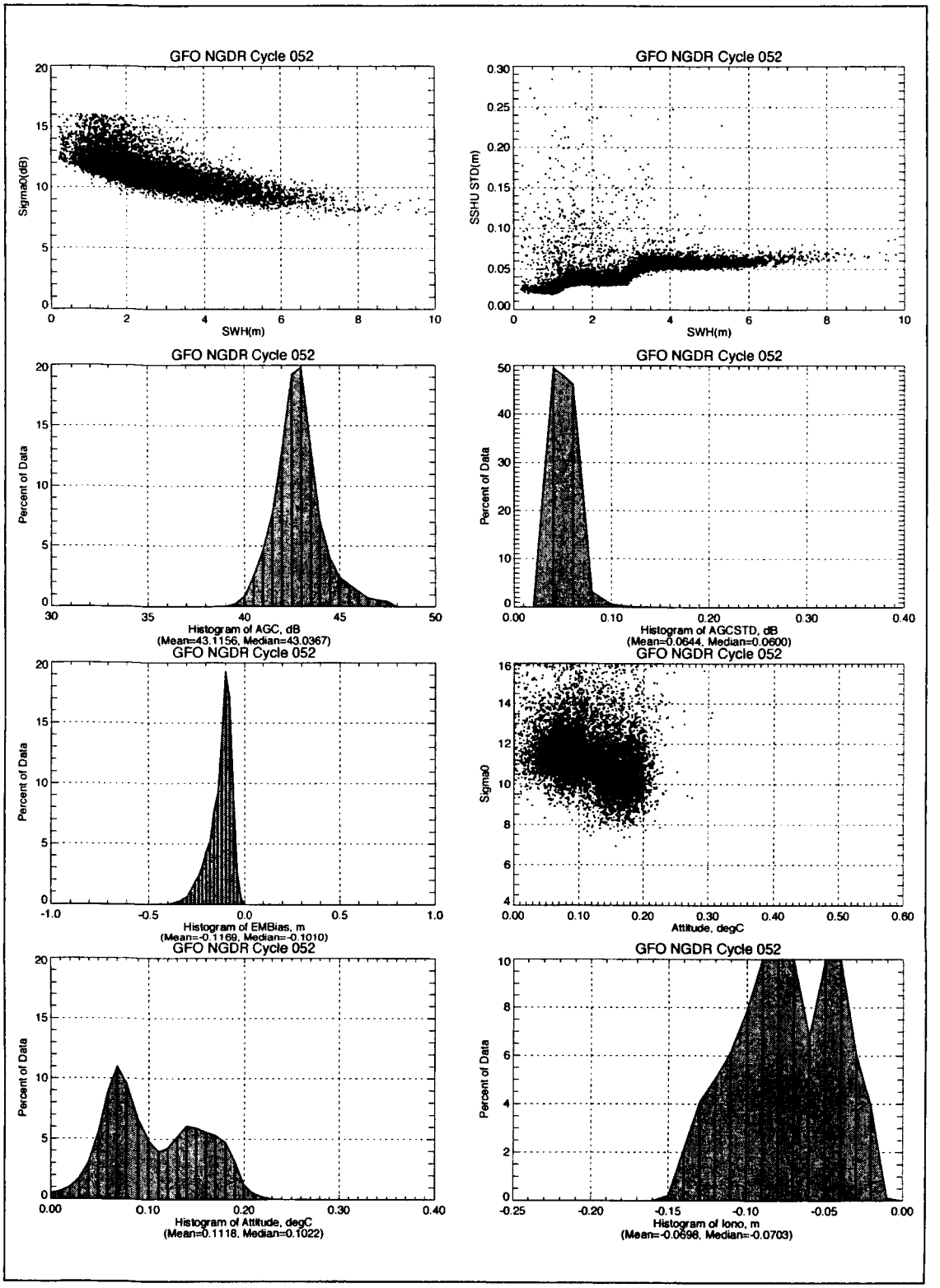

Figure B-11 NGDR History Cycle Plot (Continued)

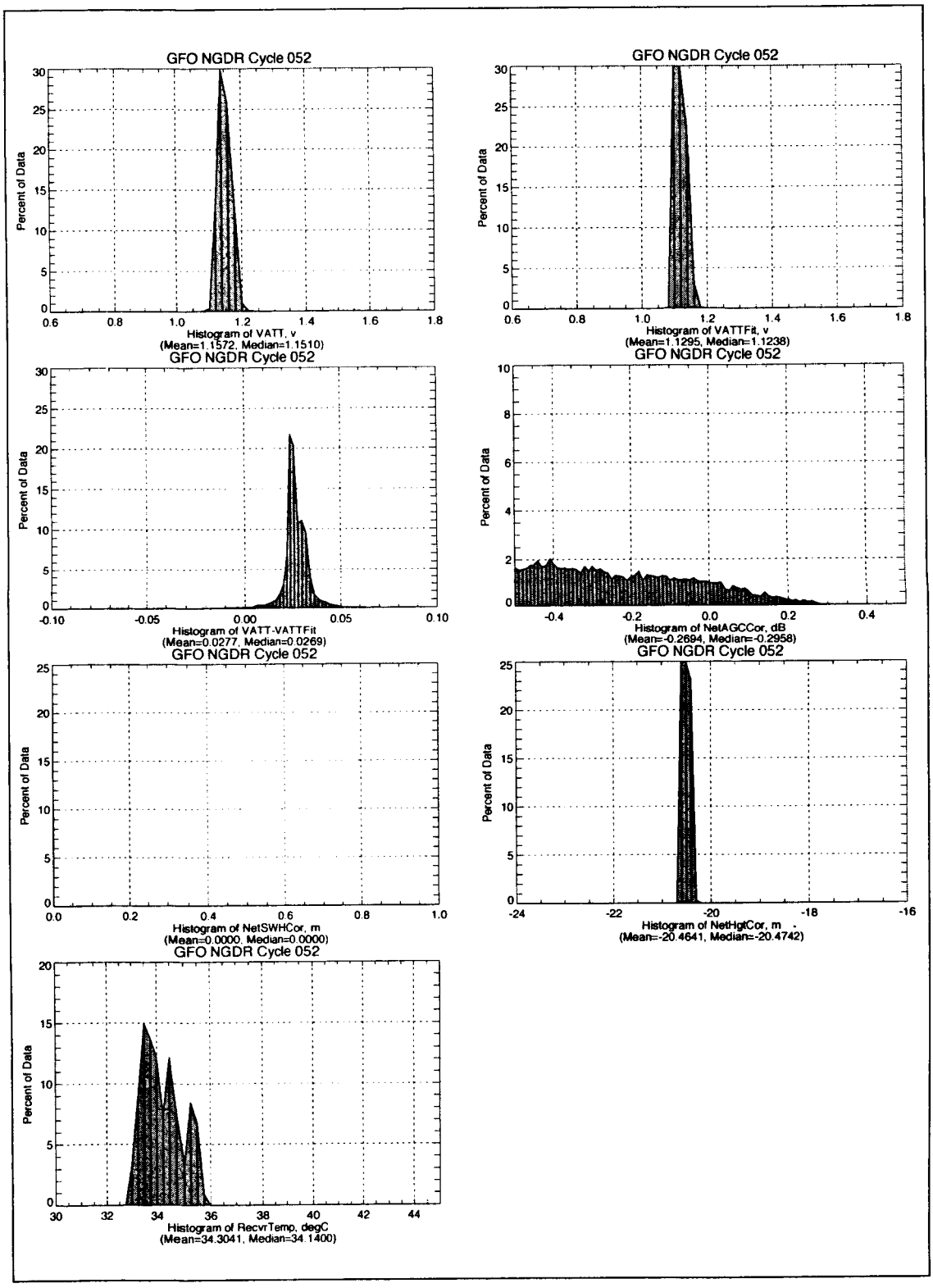

Figure B-11 NGDR History Cycle Plot (Continued)

B.12 RA Science Quicklook Plot

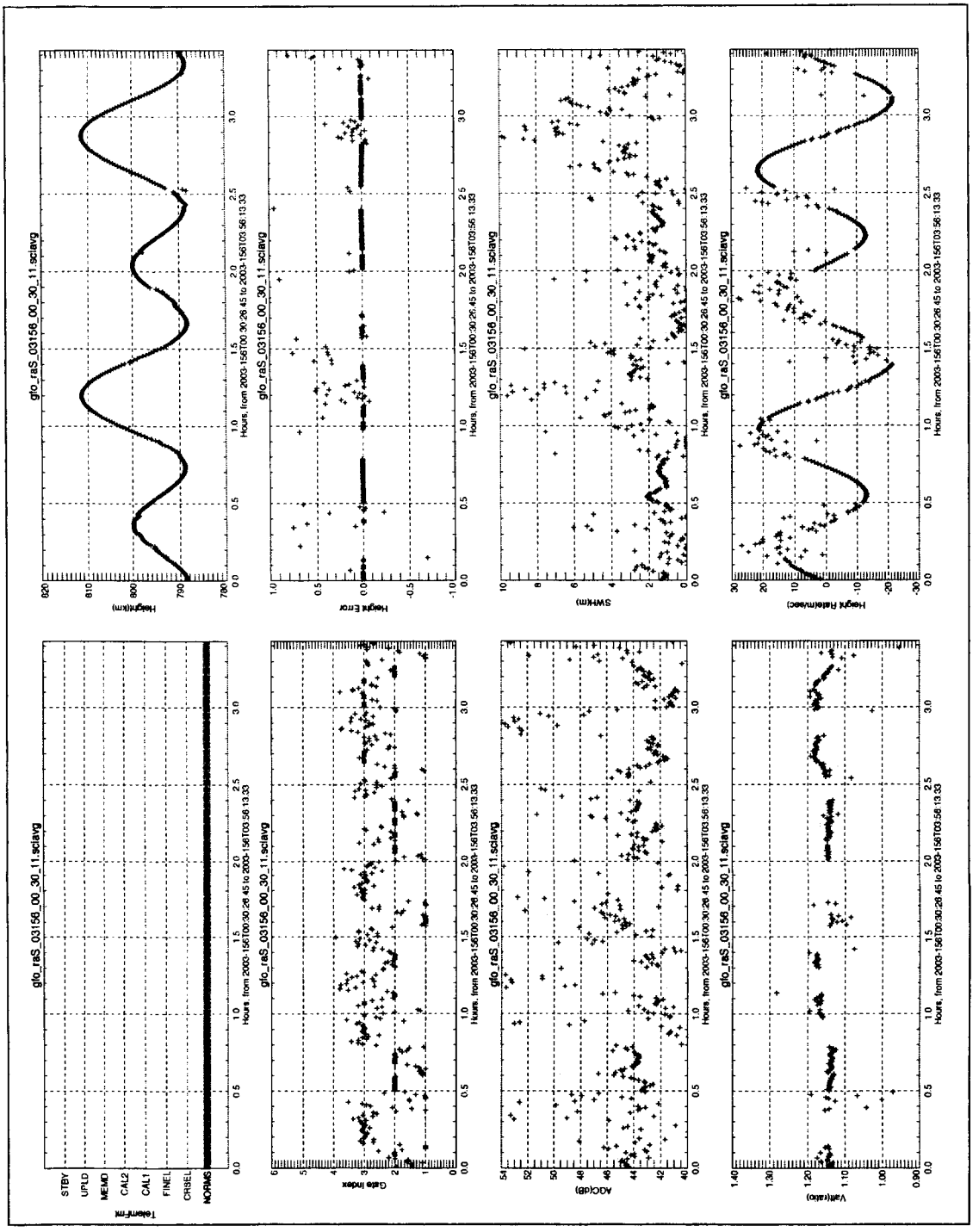

Figure B-12 RA Science Quicklook Plot

B.13 Example of a sqlldr Log File

```
SQL*Loader: Release 8.1.7.0.0 - Production on Mon Jun 2 15:31:02 2003


(c) Copyright 2000 Oracle Corporation.  All rights reserved.


Control File:   /gen/gfo/dist/bin/gfo_ra_eng.ctl
Data File:       gfoEng03156.db
Bad File:     /gen/gfo/dist/bin/gfoEng03156.bad
Discard File:  none specified


(Allow all discards)


Number to load: ALL
Number to skip: 0
Errors allowed: 50
Bind array:     64 rows, maximum of 65536 bytes
Continuation:    none specified
Path used:      Conventional
Silent options: FEEDBACK, ERRORS and DISCARDS


Table TEMP_GFO_RA_ENG, loaded from every logical record.
Insert option in effect for this table: REPLACE


Column Name                  Position   Len  Term Encl Datatype
------------------------------------------------------------------------
TIMESEC                           FIRST     *      WHT    CHARACTER
ATB                               NEXT      *      WHT    CHARACTER
STAT                              NEXT      *      WHT    CHARACTER
TESTID                            NEXT      *      WHT    CHARACTER
SEGMENT                           NEXT      *      WHT    CHARACTER
RECCOUNT                          NEXT      *      WHT    CHARACTER
ENGTEMP01                         NEXT      *      WHT    CHARACTER
ENGTEMP02                         NEXT      *      WHT    CHARACTER
ENGTEMP03                         NEXT      *      WHT    CHARACTER
ENGTEMP04                         NEXT      *      WHT    CHARACTER
ENGTEMP05                         NEXT      *      WHT    CHARACTER
ENGTEMP06                         NEXT      *      WHT    CHARACTER
ENGTEMP07                         NEXT      *      WHT    CHARACTER
ENGTEMP08                         NEXT      *      WHT    CHARACTER
ENGTEMP09                         NEXT      *      WHT    CHARACTER
ENGTEMP10                         NEXT      *      WHT    CHARACTER
ENGTEMP11                         NEXT      *      WHT    CHARACTER
```

Figure B-13 Example of a sqlldr Log File

```
ENGTEMP12                               NEXT          *       WHT     CHARACTER
COMPRECVRTEMP                           NEXT          *       WHT     CHARACTER
COLATITUDE                              NEXT          *       WHT     CHARACTER


Table TEMP_GFO_RA_ENG:
149 Rows successfully loaded.
0 Rows not loaded due to data errors.
0 Rows not loaded because all WHEN clauses were failed.
0 Rows not loaded because all fields were null.


Space allocated for bind array:                  61920 bytes(12 rows)
Space allocated for memory besides bind array:       0 bytes


Total logical records skipped:             0
Total logical records read:                149
Total logical records rejected:            0
Total logical records discarded:           0


Run began on Mon Jun 06  08:05:02 2003
Run ended on Mon Jun 06  08:06:47 2003


Elapsed time was:      00:01:45.02
CPU time was:          00:00:01.32
```

Figure B-13 Example of a sqlldr Log File (Continued)

B.14 Example of a Subset of sqlldr Log File Emailed to the GFO Administrator

```
Control File:         /gen/gfo/dist/bin/gfo_ra_eng.ctl
Data File:            gfoEng03156.db
Bad File:             /gen/gfo/dist/bin/gfoEng03156.bad
Discard File:  none specified
149 Rows successfully loaded.
0 Rows not loaded due to data errors.
0 Rows not loaded because all WHEN clauses were failed.
0 Rows not loaded because all fields were null.
Total logical records skipped:             0
Total logical records read:                149
Total logical records rejected:            0
Total logical records discarded:           0
```

Figure B-14 Example of a Subset of sqlldr Log File Emailed to the GFO Administrator

B.15 Calibration Trend

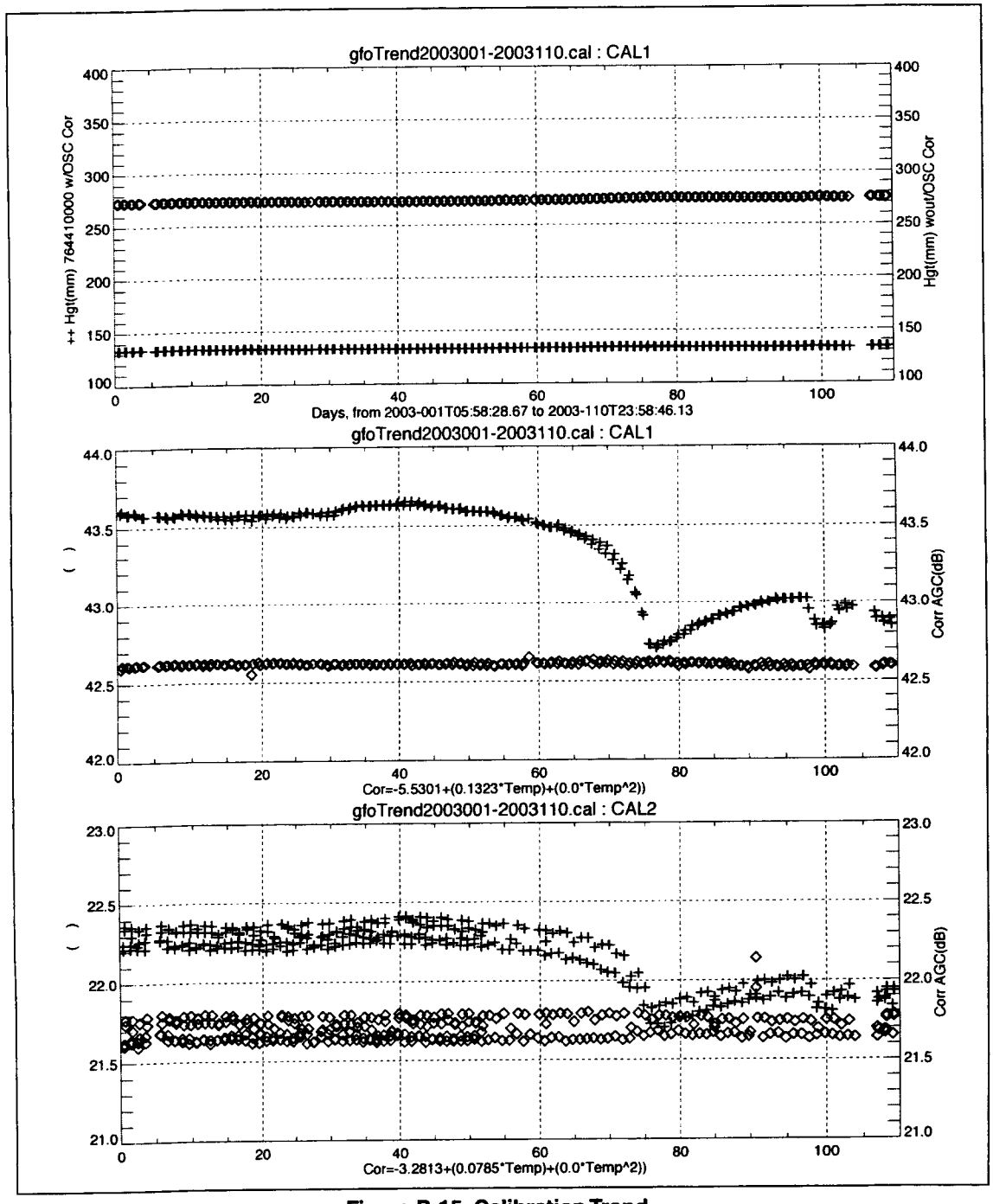

Figure B-15 Calibration Trend

B.16 RA Waveform Plots

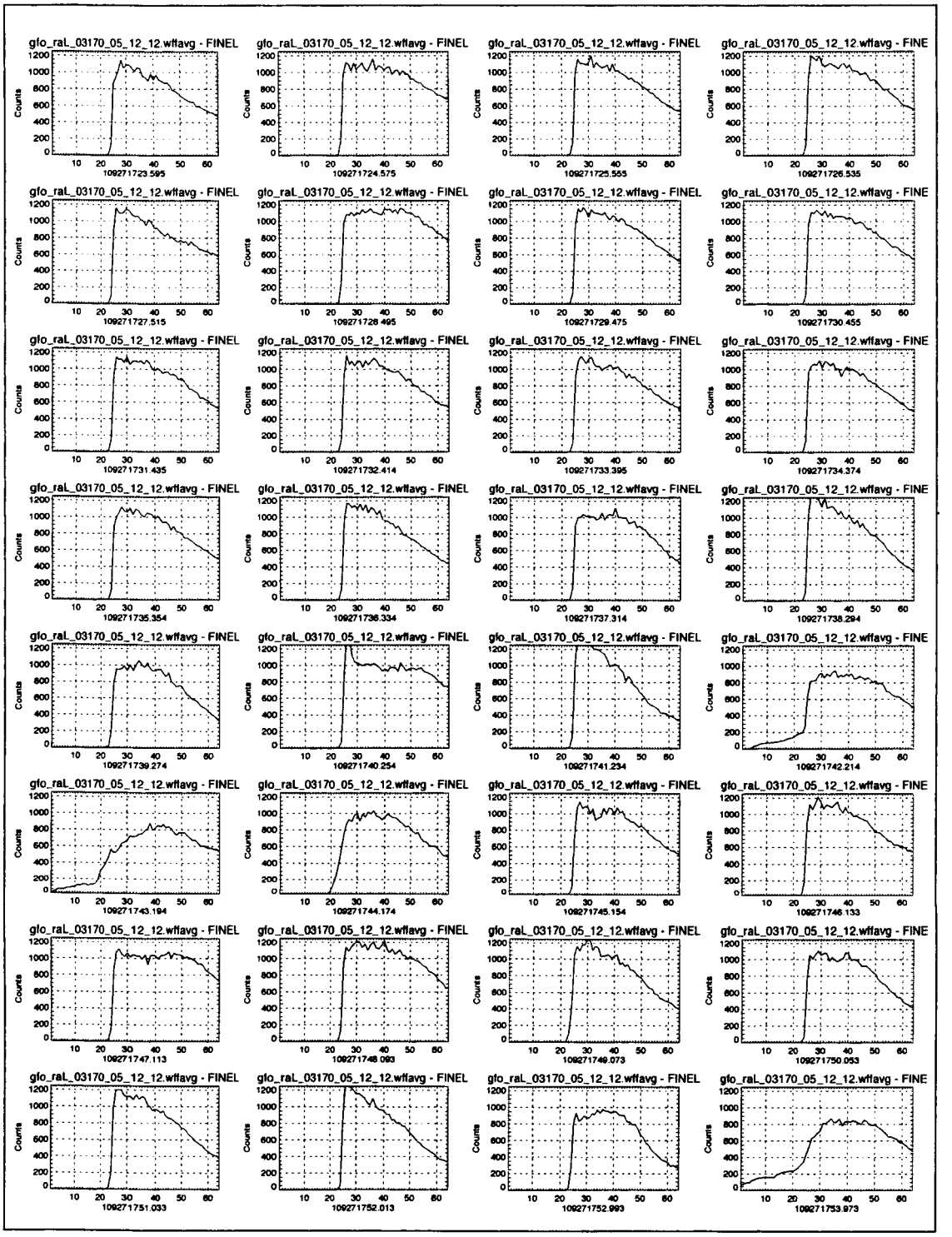

Figure B-16 RA Waveform Plots

B.17 Cycle Trend Plot

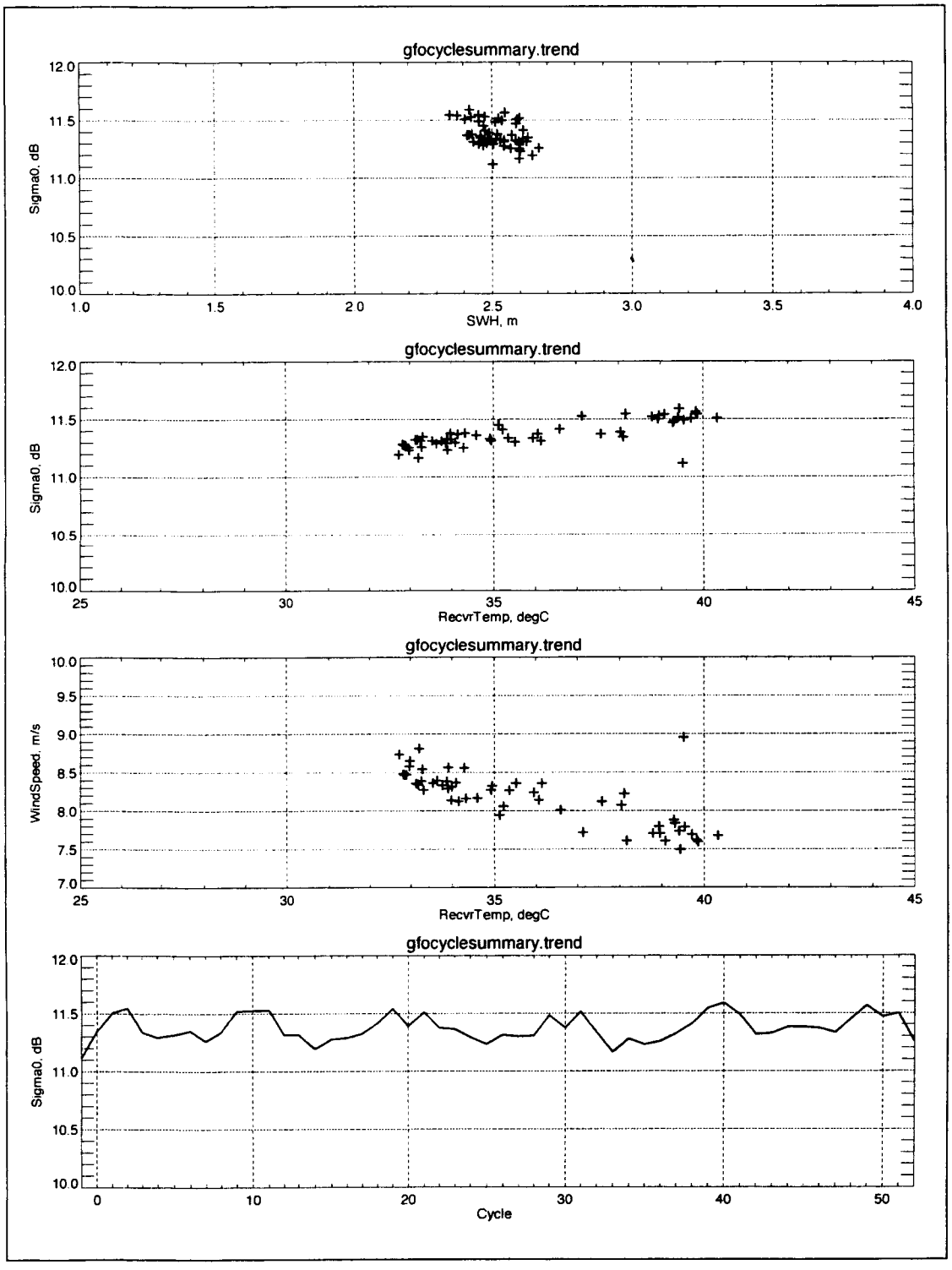

Figure B-17 Cycle Trend Plot

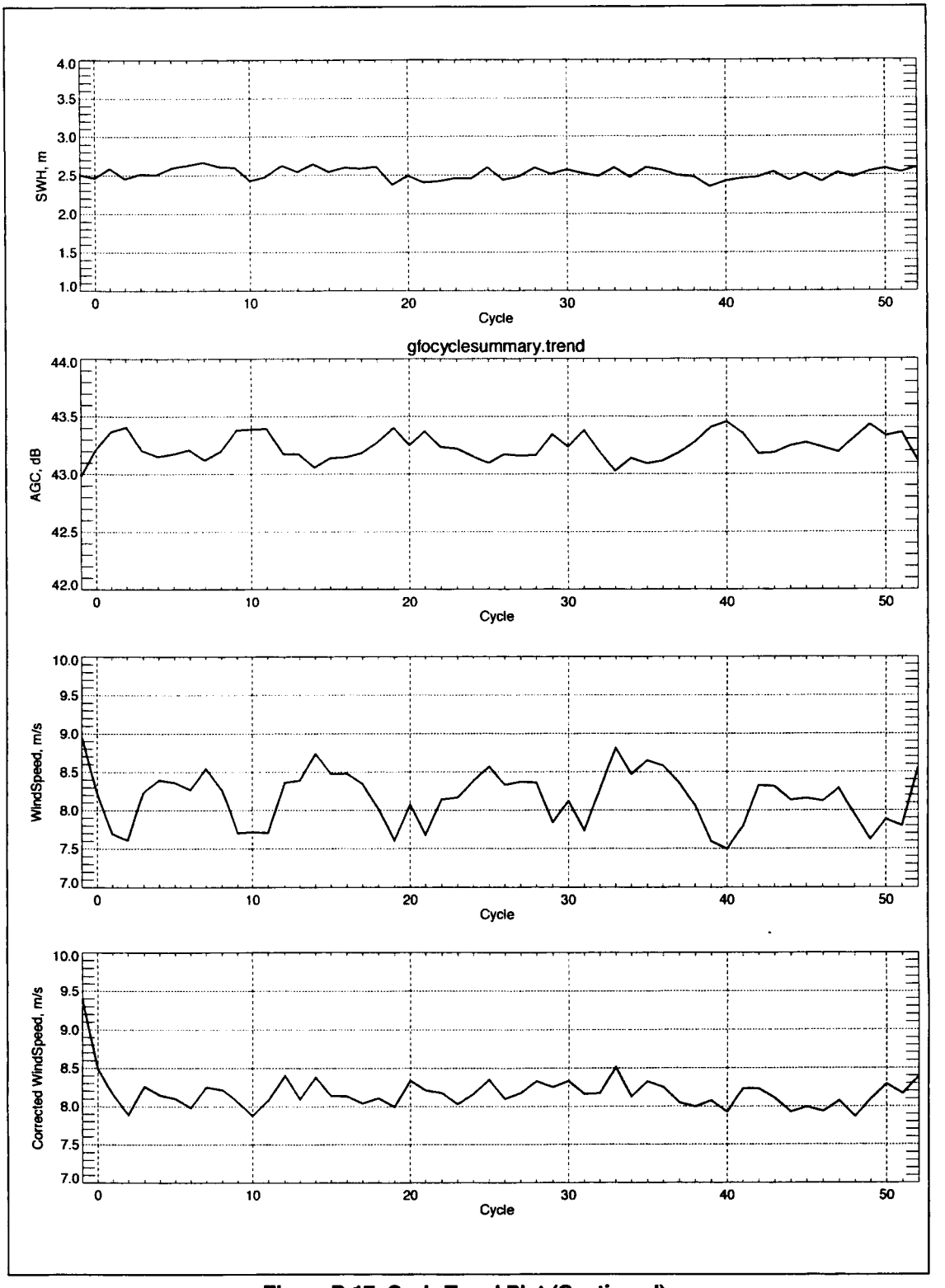

Figure B-17 Cycle Trend Plot (Continued)

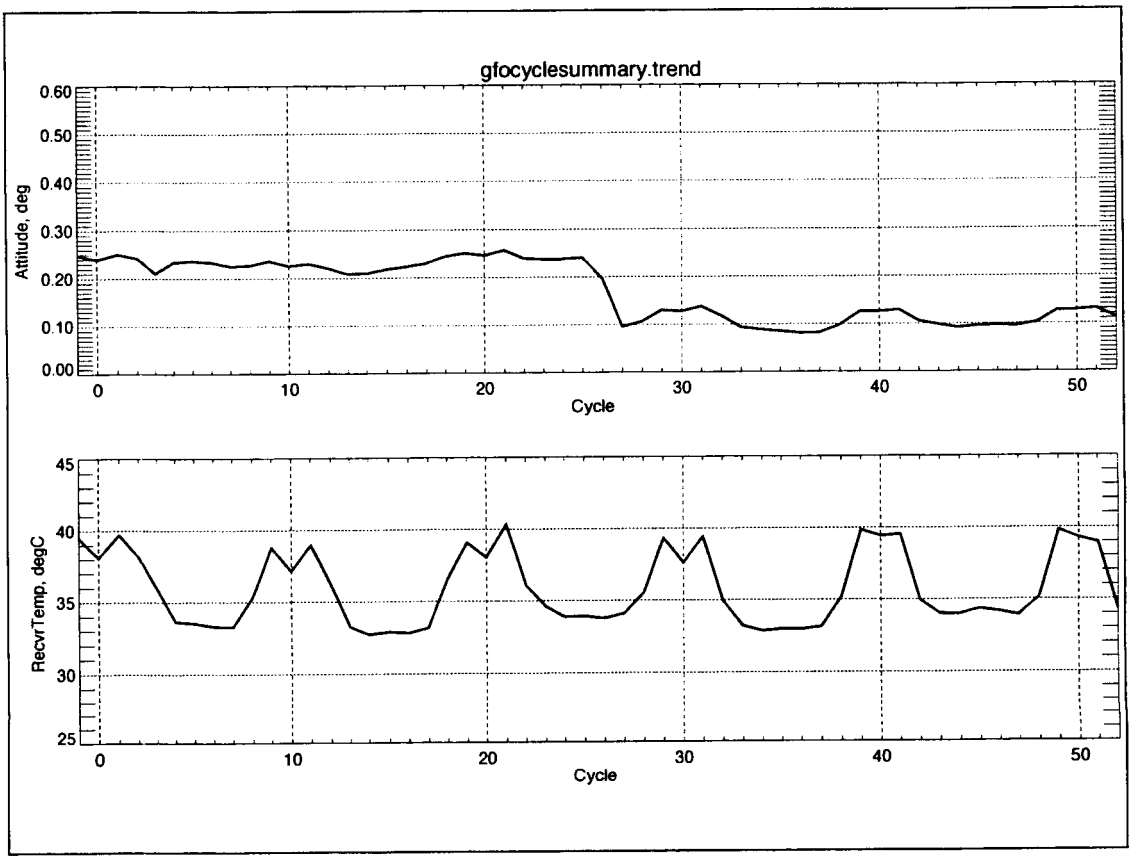

Figure B-17 Cycle Trend Plot (Continued)

B.18 NGDR Map Plot

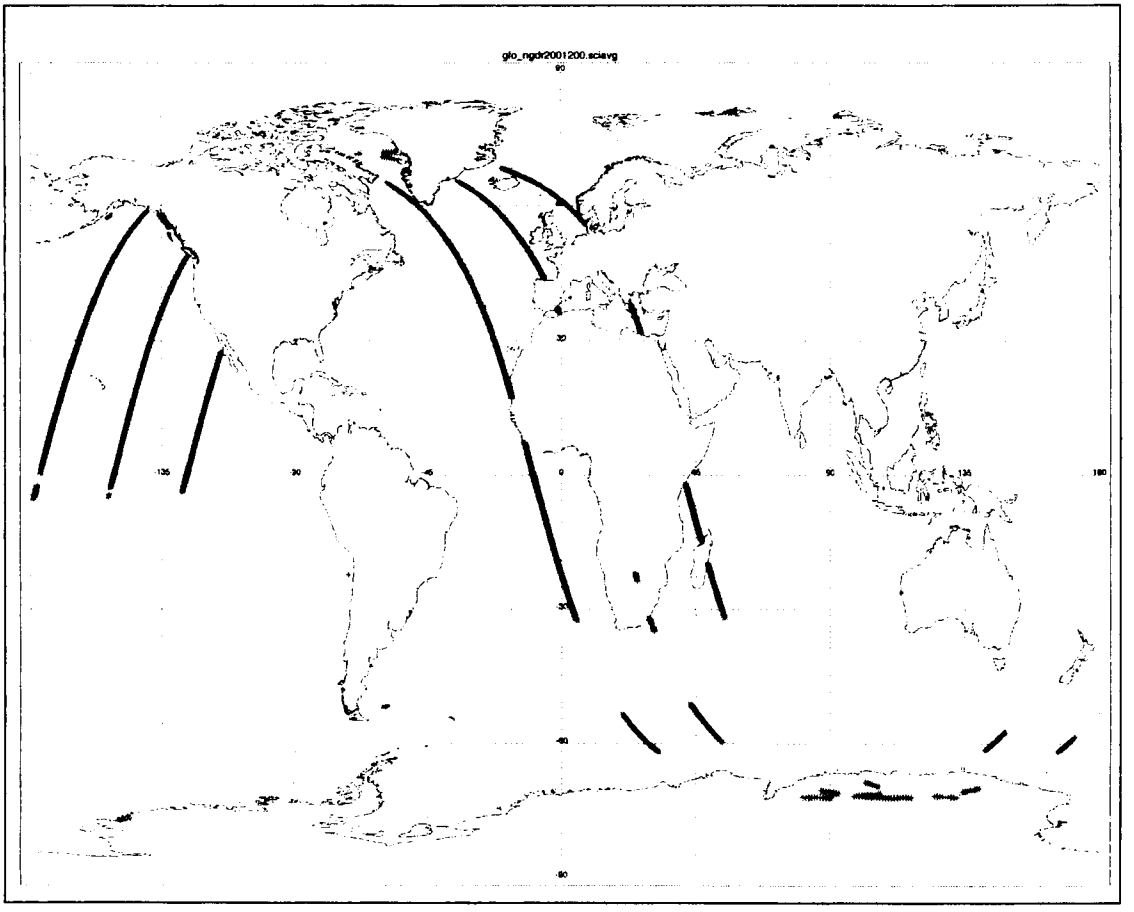

Figure B-18 NGDR Map Plot

B.19 CAL2 Waveform Gate Plot

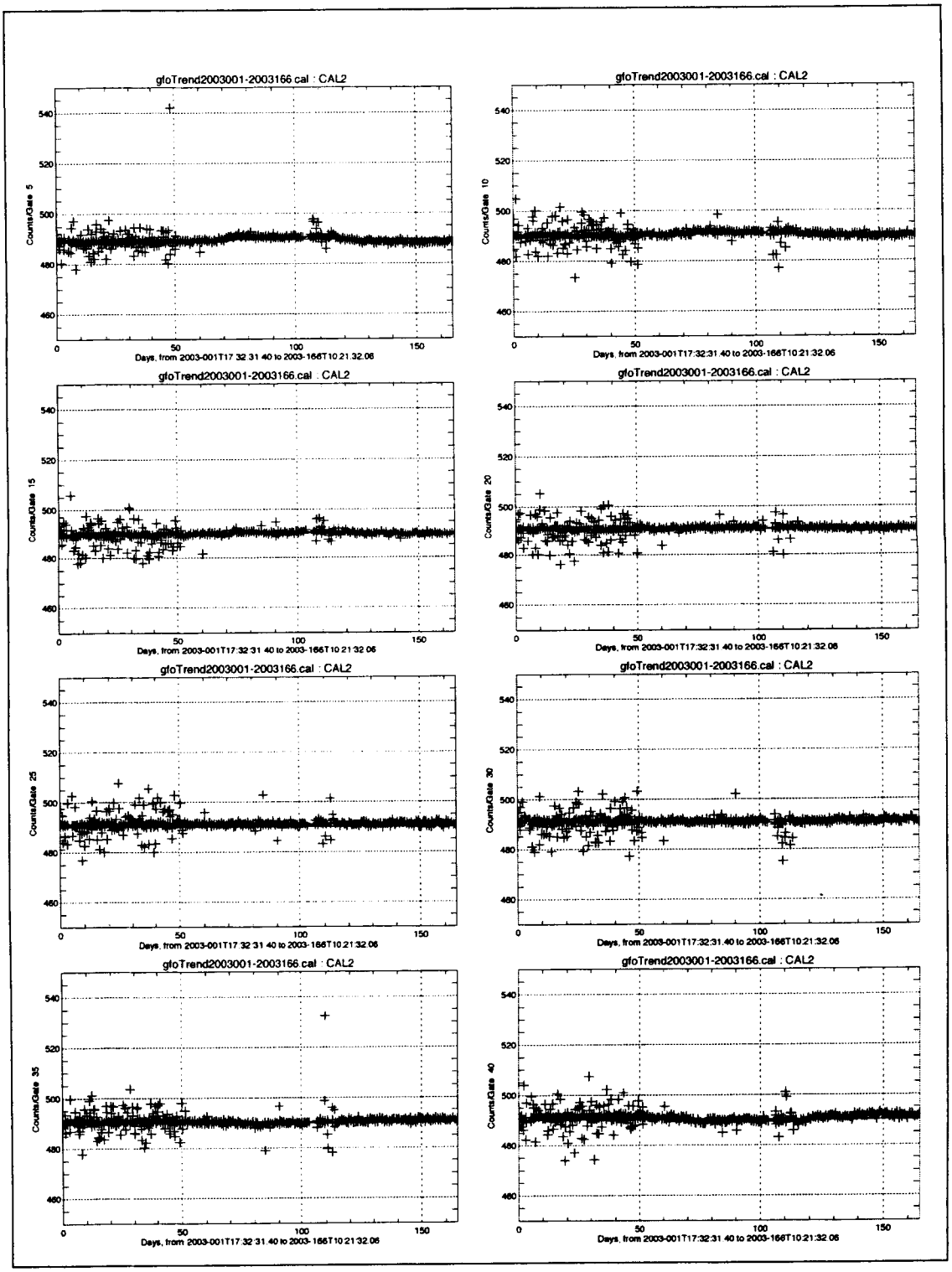

Figure B-19 CAL2 Waveform Gate Plot

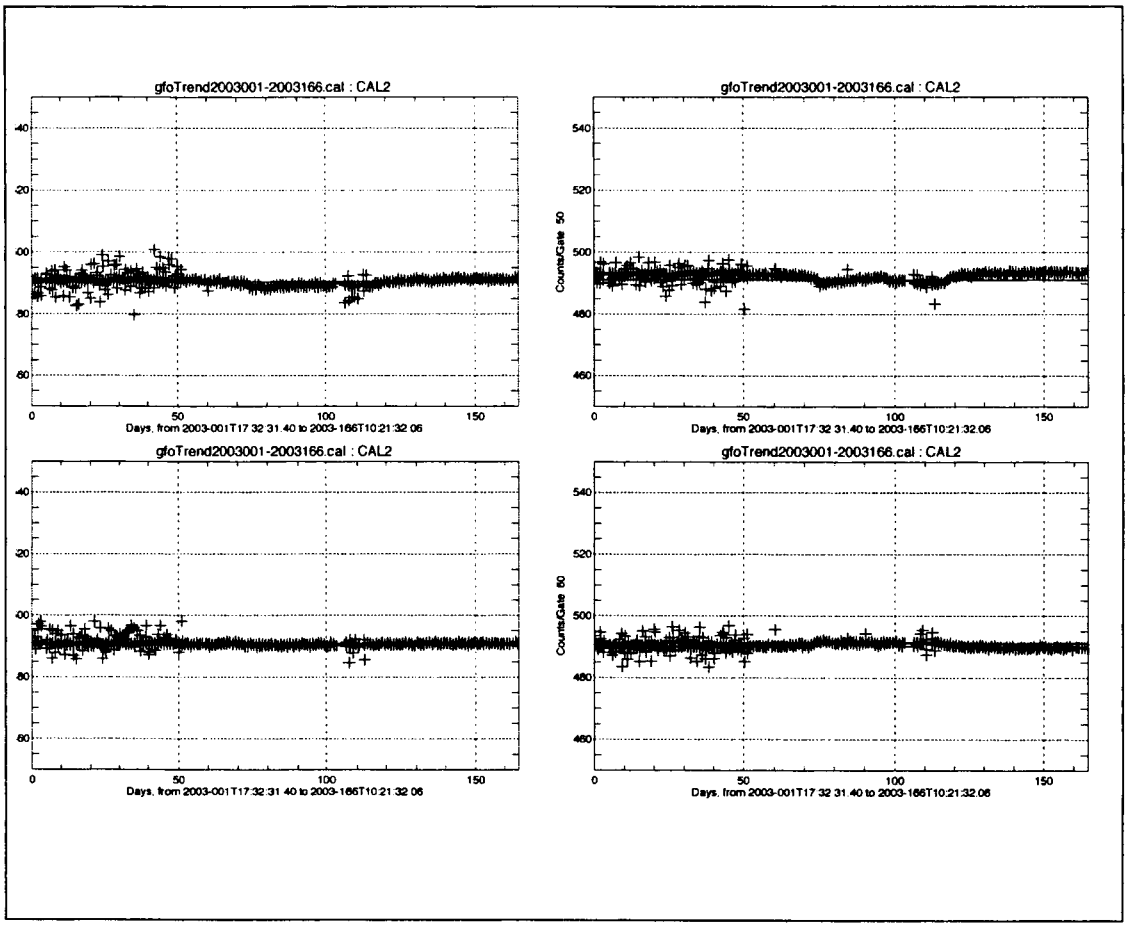

Figure B-19 CAL2 Waveform Gate Plot (Continued)

Appendix C
NGDRSciAvg Process

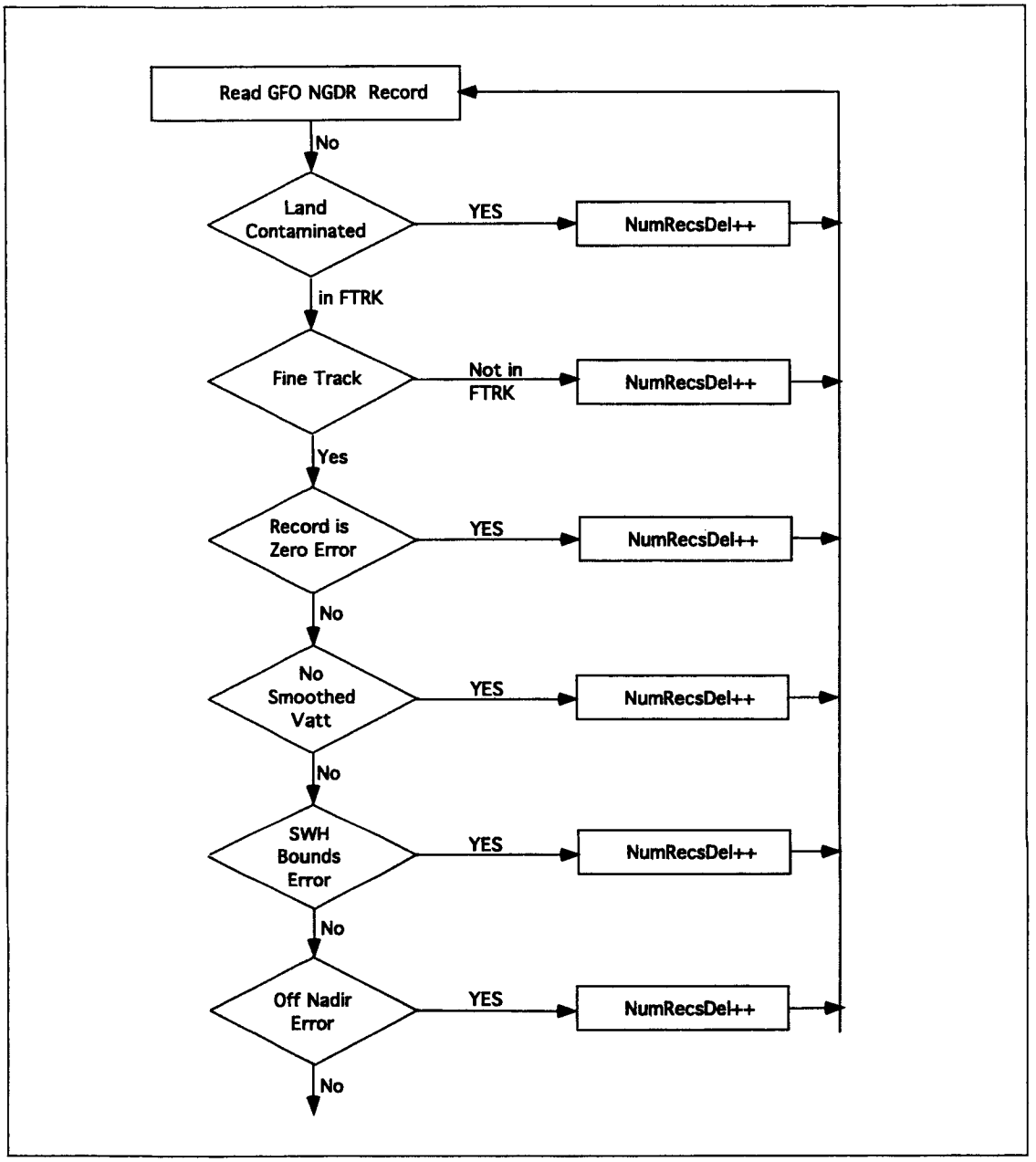

Figure C-1 NGDRSciAvg Process

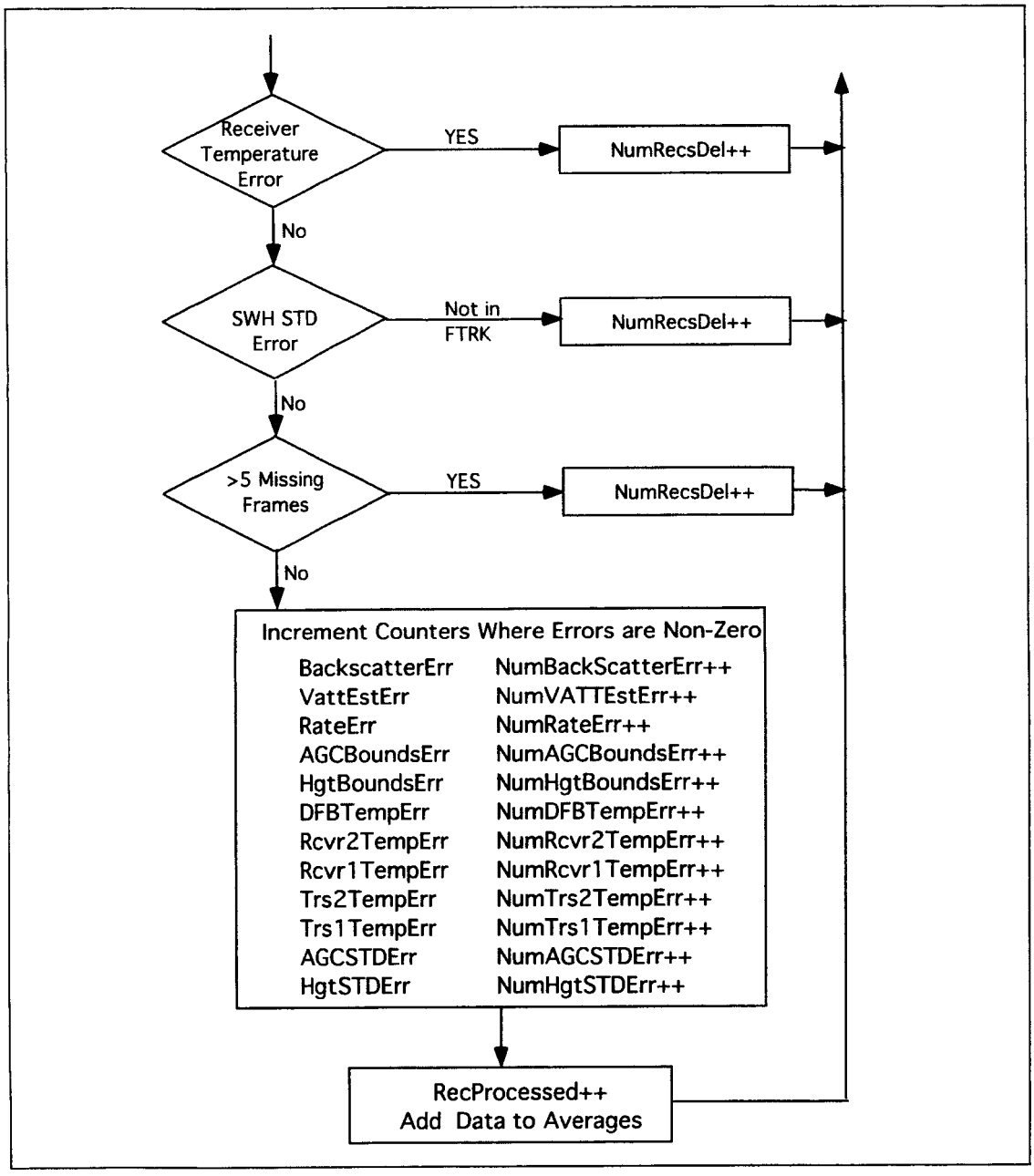

Figure C-1 NGDRSciAvg Process (Continued)

Appendix D
Output Data Type Formats

D.1 RA (Short) Science Average Format

Table D-1 RA (Short) Science Average Format

Name	Units	Fmt	Description/Computation
J2Seconds	sec	F16.2	UTC Time in seconds from J2000
ATB		A20	UTC Date and Time
Stat	text	A4	Min/Max/RMS/Mean/Full
TestID	text	A20	RA Configuration (Ex: RA1 SSPA1)
Segment	text	A20	Segment ID (filename)
RecCount	cnts	F5.0	Number of records used in computing average
TrkFmt	text	A5	Track Format (NORMS, CAL1, CAL2,etc.)
Height	m	F18.8	=(counts * Scale * SpeedLight / 2.0) + Offset Scale=819.2E-6 / 2147483648.0; Offset= 735000.0 SpeedLight=2.99792458E+8
HgtRate	m/s	F13.6	=(counts * Scale * SpeedLight / 2.0) * 1000.0d0 Scale=819.2E-6 / 68719476736.0D0; Speed-Light=2.99792458E+8
HgtError	m	F13.6	=counts * Scale* SpeedLight / 2.0; Scale=819.2E-6 / 2097152.0
AGC	dB	F13.6	=counts*Scale; Scale=64.0 / 65536.0
SWH	m	F13.6	=counts*Scale; Scale=20.0 / 256.0
Vatt	ratio	F13.6	=counts*Scale; Scale= 2.0 / 4096.0
GateIndex	index	F13.6	Gate Index
BWidth	records	F7.2	Number of records deleted by BandWidth=5MHz
TrkMode	records	F7.2	Number of records deleted by Track Mode = Acquisition
AGCSource	records	F7.2	Number of records deleted by AGCSource = SMAX
EmlHgtEr	records	F7.2	Number of records deleted by EML Hgt Error Limit = Exceeded
TypeTrack	records	F7.2	Number of records deleted by TypeTrack = Threshold
Variability	records	F7.2	Number of good records with Variability=High
TrackStatus	records	F7.2	Total number of records deleted by at least one track flag
Note: TAB delimited file			

D.2 RA Engineering Average Format

Table D-2 RA Engineering Average Format

Name	Units	Fmt	Description/Computation
J2Seconds	sec	F16.3	UTC Time in seconds from J2000
ATB	txt	A20	UTC Data and Time
Stat	txt	A4	Min/Max/RMS/Mean/Full
TestID	txt	A20	RA Configuration (Ex: RA1 SSPA1)
Segment	txt	A20	Segment ID (filename)
RecCount	counts	F5.0	Number of records in computing average
EngTemp01	degC	F9.4	RA 1 Receiver Temp
EngTemp02	degC	F9.4	RA 1 TRS 1 Temp
EngTemp03	degC	F9.4	RA 1 TRS 2 Temp
EngTemp04	degC	F9.4	RA 1 Receiver 1 Temp
EngTemp05	degC	F9.4	RA 1 Receiver 2 Temp
EngTemp06	degC	F9.4	RA 1 DFB Temp
EngTemp07	degC	F9.4	RA 2 Receiver Temp
EngTemp08	degC	F9.4	RA 2 TRS 1 Temp
EngTemp09	degC	F9.4	RA 2 TRS 2 Temp
EngTemp10	degC	F9.4	RA 2 Receiver 1 Temp
EngTemp11	degC	F9.4	RA 2 Receiver 2 Temp
EngTemp12	degC	F9.4	RA 2 DFB Temp
CompRecvrTemp	degC	F9.4	Composite Receiver Temp
CoLatitude	deg	F9.4	Co-Latitude(0-180)
Note: Tab delimited file			

D.3 RA (Long) Science with Waveforms Average Format

Table D-3 RA (Long) Science with Waveforms Average Format

Name	Units	Fmt	Description/Computation
Seconds	sec	F16.4	UTC Time in seconds from J2000
ATB	date	A20	UTC Date and Time
Latitude	id	F13.6	Latitude (-90 to +90)
Stat	id	A4	Min/Max/RMS/Mean/Full
TestID	txt	A20	RA Configuration (Ex: RA1 SSPA1)
Segment	txt	A20	Segment ID (filename)
RecCount	counts	F5.0	Number of records used in computing average
TrkFmt	txt	A5	Track Format (FINEL,CRSEL,CAL1,CAL2, etc.)
Height	meters	F18.8	Height
AGC	dB	F13.6	AGC
SWH	m	F13.6	SWH
VAT	ratio	F13.6	VATT
AGCGateAmp	counts	F13.6	Scaled counts*(Waveform Scaling Factor+1)
LEBinCnt	counts	F13.6	Scaled counts*(Waveform Scaling Factor+1)
EarlyGateAmp	counts	F13.6	Scaled counts*(Waveform Scaling Factor+1)
LateGateAmp	counts	F13.6	Scaled counts*(Waveform Scaling Factor+1)
MidGateAmp	counts	F13.6	Scaled counts*(Waveform Scaling Factor+1)
AttGateAmp	counts	F13.6	Scaled counts*(Waveform Scaling Factor+1)
NoiseGateAmp	counts	F13.6	Scaled counts*(Waveform Scaling Factor+1)
SignalNoise	counts	F13.6	Scaled counts*(Waveform Scaling Factor+1)
SubMode	counts	F13.6	Scaled counts*(Waveform Scaling Factor+1)
GateIndex	counts	F13.6	Scaled counts*(Waveform Scaling Factor+1)
Gate01	counts	F10.2	Value for Gate 1
Gate02	counts	F10.2	Value for Gate 2
.	.	.	.
.	.	.	.
.	.	.	.
Gate63	counts	F10.2	Value for Gate 63
Gate64	counts	F10.2	Value for Gate 64
Note: TAB delimited file			

D.4 RA (Long) Science Dump Format

Table D-4 RA Waveform Dump Format

Name	Units	Fmt	Description/Computation
J2Seconds	sec	F18.6	UTC Time in seconds from J2000
SciSeconds	sec	F16.8	Time in seconds of the day
TFmt	text	A5	Track Format(FINEL,CRSEL,etc.)
Status1	###	I6	Value of Status1
Status2	###	I6	Value of Status2
Height	m	F18.8	Height
HgtRate	m/s	F12.6	Height Rate
HgtError	m	F12.6	Height Error
AGC	dB	F12.6	AGC
GateFlag	#	I4	Gate Index Change Flag
GateIndex	#	I4	Gate Index
VATT	ratio	F12.6	VATT
SWH	m	F12.6	SWH
AGCGateAmp	counts	F12.6	Scaled Counts * (Waveform Scaling Factor + 1)
EarlyGateAmp	counts	F12.6	Scaled Counts * (Waveform Scaling Factor + 1)
LateGateAmp	counts	F12.6	Scaled Counts * (Waveform Scaling Factor + 1)
MidGateAmp	counts	F12.6	Scaled Counts * (Waveform Scaling Factor + 1)
LEBinCnt	counts	F12.6	Scaled Counts * (Waveform Scaling Factor + 1)
AttGateAmp	counts	F12.6	Scaled Counts * (Waveform Scaling Factor + 1)
NoiseGateAmp	counts	F12.6	Scaled Counts * (Waveform Scaling Factor + 1)
CoLatitude	deg	F9.4	Co-Latitude(0-180)
WaveScale	#	F6.2	Waveform Scaling Factor
Waveforms01	counts	F8.2	Scaled Counts * (Waveform Scaling Factor + 1)
Waveforms02	counts	F8.2	Scaled Counts * (Waveform Scaling Factor + 1)
. . .	. . .	. . .	. . .
Waveforms63	counts	F8.2	Scaled Counts * (Waveform Scaling Factor + 1)
Waveforms64	counts	F8.2	Scaled Counts * (Waveform Scaling Factor + 1)

Note: TAB delimited file

D.5 RA (Short) Science Dump Format

Table D-5 RA Science Dump Format

Name	Units	Fmt	Description/Computation
J2Seconds	sec	F18.6	UTC Time in seconds from J2000
SciSeconds	sec	F16.8	Time in seconds of the day
TFmt	text	A5	Track Format(FINEL,CRSEL,etc)
Status1	###	I6	Value of Status1
Status2	###	I6	Value of Status2
Height	m	F18.8	Height
HgtRate	m/s	F12.6	Height Rate
HgtError	m	F12.6	Height Error
AGC	dB	F12.6	AGC
GateFlag	#	I4	Gate Index Change Flag
GateIndex	#	I4	Gate Index
VATT	ratio	F12.6	VATT
SWH	m	F12.6	SWH
CoLatitude	deg	F9.4	Co-Latitude(0-180)

Note: TAB delimited file

D.6 RA Engineering Dump Format

Table D-6 RA Engineering Dump Format

Name	Units	Fmt	Description/Computation
J2Seconds	sec	F18.6	UTC Time in seconds from J2000
EngSeconds	sec	F16.8	Time in seconds of the day
RAConfig	txt	A11	RA Configuration
WVR	txt	A5	WVR Configuration
EngTemp01	degC	F9.4	RA 1 Receiver Temp
EngTemp02	degC	F9.4	RA 1 TRS 1 Temp
EngTemp03	degC	F9.4	RA 1 TRS 2 Temp
EngTemp04	degC	F9.4	RA 1 Receiver 1 Temp
EngTemp05	degC	F9.4	RA 1 Receiver 2 Temp
EngTemp06	degC	F9.4	RA 1 DFB Temp
EngTemp07	degC	F9.4	RA 2 Receiver Temp
EngTemp08	degC	F9.4	RA 2 TRS 1 Temp
EngTemp09	degC	F9.4	RA 2 TRS 2 Temp
EngTemp10	degC	F9.4	RA 2 Receiver 1 Temp
EngTemp11	degC	F9.4	RA 2 Receiver 2 Temp
EngTemp12	degC	F9.4	RA 2 DFB Temp
CompRecvrTemp	degC	F9.4	Composite Receiver Temp
CoLatitude	deg	F9.4	Co-Latitude(0-180)
Note: TAB delimited file			

D.7 SDR Dump Format

Table D-7 SDR Dump Format

Name	Units	Fmt	Description/Computation
SDRJ2Seconds	sec	F18.6	J2000 UTC Seconds
FrameUTC	sec	F16.8	Frame Seconds
RAStatusMode1	#	I12	RA Status Mode I
RAStatusMode2	#	I12	RA Status Mode II
QualityWord1	#	I12	RA Quality Test Results
QualityWord2	#	I12	WVR Quality Test Results
GateIndex(01)	#	I4	Gate Index for RA Frame 1
GateIndex(02)	#	I4	Gate Index for RA Frame 2
GateIndex(03)	#	I4	Gate Index for RA Frame 3
GateIndex(04)	#	I4	Gate Index for RA Frame 4
GateIndex(05)	#	I4	Gate Index for RA Frame 5
GateIndex(06)	#	I4	Gate Index for RA Frame 6
GateIndex(07)	#	I4	Gate Index for RA Frame 7
GateIndex(08)	#	I4	Gate Index for RA Frame 8
GateIndex(09)	#	I4	Gate Index for RA Frame 9
GateIndex(10)	#	I4	Gate Index for RA Frame 10
HgtWord(01)	mm	F18.8	Sea Surface Height for RA Frame 1
HgtWord(02)	mm	F18.8	Sea Surface Height for RA Frame 2
HgtWord(03)	mm	F18.8	Sea Surface Height for RA Frame 3
HgtWord(04)	mm	F18.8	Sea Surface Height for RA Frame 4
HgtWord(05)	mm	F18.8	Sea Surface Height for RA Frame 5
HgtWord(06)	mm	F18.8	Sea Surface Height for RA Frame 6
HgtWord(07)	mm	F18.8	Sea Surface Height for RA Frame 7
HgtWord(08)	mm	F18.8	Sea Surface Height for RA Frame 8
HgtWord(09)	mm	F18.8	Sea Surface Height for RA Frame 9
HgtWord(10)	mm	F18.8	Sea Surface Height for RA Frame 10
AvgHgtWord	mm	F18.8	Averaged 10/sec Frame
HeightRate	m/sec	F16.6	Height Rate
HgtWordSTD	mm	F16.6	Height Word Standard Deviation
FMCrosstalk	mm	F16.6	FM Crosstalk

Table D-7 SDR Dump Format (Continued)

Name	Units	Fmt	Description/Computation
SWH(01)	m	F12.6	SWH for RA Frame 1
SWH(02)	m	F12.6	SWH for RA Frame 2
SWH(03)	m	F12.6	SWH for RA Frame 3
SWH(04)	m	F12.6	SWH for RA Frame 4
SWH(05)	m	F12.6	SWH for RA Frame 5
SWH(06)	m	F12.6	SWH for RA Frame 6
SWH(07)	m	F12.6	SWH for RA Frame 7
SWH(08)	m	F12.6	SWH for RA Frame 8
SWH(09)	m	F12.6	SWH for RA Frame 9
SWH(10)	m	F12.6	SWH for RA Frame 10
AvgSWH	m	F12.6	Averaged 10/sec Frame
SWHSTD	m	F12.6	SWH Standard Deviation
SWHONABias	m	F12.6	SWH Bias
AGC(01)	dB	F12.6	AGC for RA Frame
AGC(02)	dB	F12.6	AGC for RA Frame
AGC(03)	dB	F12.6	AGC for RA Frame
AGC(04)	dB	F12.6	AGC for RA Frame
AGC(05)	dB	F12.6	AGC for RA Frame
AGC(06)	dB	F12.6	AGC for RA Frame
AGC(07)	dB	F12.6	AGC for RA Frame
AGC(08)	dB	F12.6	AGC for RA Frame
AGC(09)	dB	F12.6	AGC for RA Frame
AGC(10)	dB	F12.6	AGC for RA Frame
AvgAGC	dB	F12.6	Average 10/sec Frame
AGCSTD	dB	F12.6	AGC Standard Deviation
AGCTempCor	dB	F12.6	AGC Correction for Receiver Gain Changes
DeltaAGCHgt	dB	F12.6	Delta Height Correction for AGC
AGCCor	dB	F12.6	AGC Correction for Attitude
AttitudeBias	mm	F12.6	Attitude Wave Height Bias
OffNadir	deg	F12.6	Off-nadir Angle
BackScatter	dB	F12.6	Backscatter Coefficient
PathDelay	cm	F12.6	Altimeter Path Delay

Table D-7 SDR Dump Format (Continued)

Name	Units	Fmt	Description/Computation
BrightTemp22	deg K	F12.6	22 GHz Brightness Temperature
BrightTemp37	deg K	F12.6	32 GHz Brightness Temperature
AvgVatt	v	F12.6	Voltage Proportional to Attitude
FittedVatt	v	F12.6	Fit to the VATT
RecvrTemp	deg C	F12.6	Receiver Temperature
Note: TAB delimited file			

D.8 NGDR Dump Format

Table D-8 NGDR Dump Format

Name	Units	Fmt	Description/Computation
J2Seconds	sec	F18.6	Converted to J2000 Epoch
FrameUTC	sec	F18.6	Frame Time in Seconds
StatusMode1	bits	I12	RA Status Mode I
StatusMode2	bits	I12	RA Status Mode II
QualityWord1	bits	I12	RA Quality Test Results
QualityWord2	bits	I12	WVR Quality Test Results
Flags1	bits	I12	Flag
Flags2	bits	I12	Flag
InstState	bits	I12	Instrument Stat Flag
Latitude	deg	F12.6	Latitude
Longitude	deg	F12.6	East Longitude
SSHUnCor	mm	F18.6	Sea Surface Height Uncorrected
SSHCor	mm	F18.6	Sea Surface Height Corrected
SatAlt	mm	F18.6	Satellite Height
TimeMDF	sec	F12.6	Time Shift Midframe
SWH	0.01m	F12.6	Significant Wave Height
Sigma0	0.01dB	F12.6	Sigma0
WindSpeed	0.1m/sec	F12.6	Wind Speed
AGC	0.01dB	F12.6	Automatic Gain Control
DryTropo	mm	F12.6	Dry Troposphere
WetTropo	mm	F12.6	Wet Troposphere
Iono	mm	F12.6	Ionosphere
InvBaro	mm	F12.6	Inverse Barometer
EMBias	mm	F12.6	EM Bias
EarthTide	mm	F12.6	Solid Earth Tide
OceanTide	mm	F12.6	Ocean Water Tide
LoadTide	mm	F12.6	Ocean Load Tide
PoleTide	mm	F12.6	Pole Tide
WaterDepth	m	F12.6	Water Depth
Geoid	m	F18.6	Geoid Height

Table D-8 NGDR Dump Format (Continued)

Name	Units	Fmt	Description/Computation
MeanSSH1	mm	F18.6	Mean Sea Surface I
MeanSSH2	mm	F18.6	Mean Sea Surface II
SSHUSTD	mm	F12.6	Sea Surface Height Uncorrected STD
SWHSTD	cm	F12.6	SWH STD
AGCSTD	0.01dB	F12.6	AGC STD
NetHgtCor	mm	F12.6	Net Height Correction
NetSWHCor	mm	F12.6	Net SWH Correction
NetAGCCor	0.01dB	F12.6	Net AGC Correction
TimeTagCor	microsec	F12.6	Net Time Tag Correction
Attitude	0.01 deg	F12.6	Attitude
NvalsSSHU	counts	F6.2	Number of Values Sea Surface Height Uncorrected Used
NvalsSWH	counts	F6.2	Number of Values SWH Used
NvalsAGC	counts	F6.2	Number of Values AGC Used
SWHHiRate(01)	0.01m	F12.6	10/sec SWH High Rate 01
SWHHiRate(02)	0.01m	F12.6	10/sec SWH High Rate 02
SWHHiRate(03)	0.01m	F12.6	10/sec SWH High Rate 03
SWHHiRate(04)	0.01m	F12.6	10/sec SWH High Rate 04
SWHHiRate(05)	0.01m	F12.6	10/sec SWH High Rate 05
SWHHiRate(06)	0.01m	F12.6	10/sec SWH High Rate 06
SWHHiRate(07)	0.01m	F12.6	10/sec SWH High Rate 07
SWHHiRate(08)	0.01m	F12.6	10/sec SWH High Rate 08
SWHHiRate(09)	0.01m	F12.6	10/sec SWH High Rate 09
SWHHiRate(10)	0.01m	F12.6	10/sec SWH High Rate 10
SSHUHiRateDif(01)	mm	F12.6	10/sec Uncorrected High Rate Height Deviations 01
SSHUHiRateDif(02)	mm	F12.6	10/sec Uncorrected High Rate Height Deviations 02
SSHUHiRateDif(03)	mm	F12.6	10/sec Uncorrected High Rate Height Deviations 03
SSHUHiRateDif(04)	mm	F12.6	10/sec Uncorrected High Rate Height Deviations 04
SSHUHiRateDif(05)	mm	F12.6	10/sec Uncorrected High Rate Height Deviations 05
SSHUHiRateDif(06)	mm	F12.6	10/sec Uncorrected High Rate Height Deviations 06
SSHUHiRateDif(07)	mm	F12.6	10/sec Uncorrected High Rate Height Deviations 07
SSHUHiRateDif(08)	mm	F12.6	10/sec Uncorrected High Rate Height Deviations 08
SSHUHiRateDif(09)	mm	F12.6	10/sec Uncorrected High Rate Height Deviations 09

Table D-8 NGDR Dump Format (Continued)

Name	Units	Fmt	Description/Computation
SSHUHiRateDif(10)	mm	F12.6	10/sec Uncorrected High Rate Height Deviations 10
SatAltHiRateDif(01)	mm	F12.6	10/sec High Rate Orbit Deviations 01
SatAltHiRateDif(02)	mm	F12.6	10/sec High Rate Orbit Deviations 02
SatAltHiRateDif(03)	mm	F12.6	10/sec High Rate Orbit Deviations 03
SatAltHiRateDif(04)	mm	F12.6	10/sec High Rate Orbit Deviations 04
SatAltHiRateDif(05)	mm	F12.6	10/sec High Rate Orbit Deviations 05
SatAltHiRateDif(06)	mm	F12.6	10/sec High Rate Orbit Deviations 06
SatAltHiRateDif(07)	mm	F12.6	10/sec High Rate Orbit Deviations 07
SatAltHiRateDif(08)	mm	F12.6	10/sec High Rate Orbit Deviations 08
SatAltHiRateDif(09)	mm	F12.6	10/sec High Rate Orbit Deviations 09
SatAltHiRateDif(10)	mm	F12.6	10/sec High Rate Orbit Deviations 10
TB22	0.01 degK	F12.6	22 GHz Brightness Temp
TB37	0.01 degK	F12.6	37 GHz Brightness Temp
RecvrTemp	0.01 degC	F12.6	Receiver Temperature
Vatt	millivolt	F12.6	Averaged VATT
VattFit	millivolt	F12.6	Fitted VATT
Note: TAB delimited file			

D.9 NGDR Header Database Format

Table D-9 NGDR Header Database Format

Field	Name	Units	Format	Description
1	Cycle	#	I3	Cycle Number (Cycle= Exactly 17 days)
2	YrDOYr	#	A7	Year and Day-of-Year of the Data
3	FileTime	char	A11	Start and Stop Time of Data in Seconds
4	DateRun	char	A8	Date Data Processed
5	CHgtCalibBias	mm	A12	Height Calibration Bias
6	CAltBiasInit	mm	A12	Altitude Bias Initial Correction
7	CAltBiasCG	mm	A12	Altitude Bias Center of Gravity
8	CSWHBiasInit	mm	A12	SWH Bias Initial
9	CAGCCalibBias	dB	A12	AGC Calibration Bias
10	CAGCBiasInit	dB	A12	AGC Bias Initial
11	TotFrames	#	I6	Total Number of Frames for day-of-year
12	TotUsed	#	I6	Total Number of Frames used for day-of-year
13	TotDelete	#	I6	Total Number of Frames deleted for day-of-year
14	TotTrkFrames	#	I6	Total Number of Frames in Fine Track
15	KwORB	char	A10	Keyword ORB for Altitude
16	KwTID	char	A10	Keyword TID for Ocean Water Tide
17	KwION	char	A10	Keyword ION for Ionosphere
18	KwDRY	char	A10	Keyword DRY for Dry Troposphere
19	KwWET	char	A10	Keyword WET for Wet Troposphere

Note: SPACE delimited file

D.10 NGDR Science Database Format

Table D-10 NGDR Science Database Format

Field	Name	Units	Format	Description
1	TimeSec	sec	F16.3	Converted to 2000 Epoch
2	ATB	date	A24	UTC Time
3	RecCount	counts	F5.1	Number Frames used in 60 sec Avg
4	TelemFmt	text	A5	Telemetry Format (NORMS, FINEL)
5	NotinFTRK	#	I4	Value: 0=Not in FTRK, 1=In FTRK
6	SSHCorrected	m	F16.6	SSH=SSHU-Environmental Corrections
7	SSHUncorrected	char	I4	Telemetry Format
8	SSHUSTD	m	F16.6	Standard Deviation from Fit applied to SSHU Values
9	Windspeed	cm/sec	F16.6	Wind Speed from Sigma0
10	SWH	m	F16.6	Significant Wave Height
11	SWHSTD	m	F16.6	Standard Deviation from Fit applied to SWH Values
12	Sigma0	dB	F16.6	Backscatter Coefficient
13	AGC	dB	F16.6	Automatic Gain Control
14	AGCSTD	dB	F16.6	Standard Deviation from Fit applied to AGC Values
15	DryTropo	m	F16.6	Dry Troposphere
16	WetTropo	m	F16.6	Wet Troposphere
17	NetAGCCorr	dB	F16.6	Net AGC Correction
18	Attitude	deg	F16.6	OffNdir Angle - Hgt above Ellipsoid
19	Iono	m	F16.6	-1*Alt Rng Corr
20	NetSWHCorr	m	F16.6	Net SWH Correction
21	InvBaro	m	F16.6	Inverse Barometer
22	Btemp22	deg K	F16.6	22 GHz Brightness Temperature
23	Btemp37	deg K	F16.6	37 GHz Brightness Temperature
24	VATT	V	F16.6	Average VATT
25	VATTFit	V	F16.6	Fitted VATT
26	RecvrTemp	deg C	F16.6	Receiver Temperature
27	Latitude	deg	F16.6	Geodetic Latitude +90N to -90S
28	Longitude	deg	F16.6	East Geodetic Longitude 0 to 360
29	EMBIAS	m	F16.6	Sea State Bias
30	NetHgtCorr	m	F16.6	Net Height Correction

Table D-10 NGDR Science Database Format (Continued)

Field	Name	Units	Format	Description
31	EarthTide	m	F16.6	Solid Earth Tide
32	Ocean Tide	m	F16.6	Ocean Water Tide
33	LoadTide	m	F16.6	Ocean Load Tide
34	PoleTide	m	F16.6	Pole Tide
35	WaterDepth	m	F16.6	Water Depth
36	Geoid	m	F16.6	Geoid Height
37	SubMode	#	F16.6	SubMode = ((4*TrkMode(1)) + (2*TypeTrack(1)) + (AGCSource(1)) + (4*TrkMode(2)) + (2*TypeTrack(2)) + (AGCSource(2))) / 2
Note: SPACE delimited file				

D.11 SDR Science Average Format

Table D-11 SDR Science Average Format

Name	Units	Format	Description
TimeSec	sec	F16.3	Converted to 2000 Epoch
ATB	date	A24	UTC Date and Time
RecCount	counts	F5.1	Number Frames used in Average
NotinFTRK	#	I4	Value 0 = Not in Fine Track, 1 = in Fine Track
AvgHgtWord	mm	F16.6	10 per sec Averaged height Word
HgtRate	m/s	F16.6	Averaged Height Rate
HgtWordSTD	mm	F16.6	Averaged Height Word STD
FMCrosstalk	mm	F16.6	Averaged FM Crosstalk
AvgSWH	m	F16.6	10 per sec Averaged SWH
SWHSTD	m	F16.6	Averaged SWH STD
SWHONABias	m	F16.6	Averaged SWH Bias
AvgAGC	dB	F16.6	10 per sec Averaged AGC
AGCSTD	dB	F16.6	Averraged AGC STD
AGCTempCor	dB	F16.6	Averaged AGC Temperature Corection
DeltaAGCHgt	dB	F16.6	Averaged Delta AGC Height
AGCCor	dB	F16.6	Averaged AGC Correction for Attitude
Attitude Bias	mm	F16.6	Averaged Attitude Wave Height Bias
OffNadir	deg	F16.6	Averged Off-nadir Angle
BackScatter	dB	F16.6	Averaged Backscatter Coefficient
PathDelay	cm	F16.6	Averaged Path Delay
BTemp22	deg K	F16.6	Averaged 22 GH_Z Brightness Temperature
BTemp37	deg K	F16.6	Averaged 37 GH_Z Brightness Temperature
AvgVAT	V	F16.6	Averaged Average VAT
FittedVatt	V	F16.6	Averaged Fitted VATT
RecvrTemp	deg C	F16.6	Averaged Receiver Temperature
AvgGateIndex	#	F16.6	Averaged Gate Index

Table D-11 SDR Science Average Format (Continued)

Name	Units	Format	Description
Submode	#	F16.6	SubMode = ((4*TrkMode(1)) + (2*TypeTrack(1)) + (AGCSource(1)) + (4*TrkMode(2)) + (2*TypeTrack(2)) + (AGCSource(2))) / 2

Note: TAB delimited file

D.12 SDR Oscillator Drift Database Format

Table D-12 SDR Oscillator Drift Database Format

	Field	Units	Format	Description
	Segment	txt	A30	Segment ID
	StartUTC	sec	F16.9	Start UTC
	UTCYear	year	I4	UTC Year
	UTCDOY	day	I3	UTC Day of Year
	UTCSeconds	sec	F16.9	UTC Seconds
	VTCWValue	ticks	F18.2	VTCW Value
	Ratio	sec/tick	F26.22	Ratio

Note: SPACE delimited file

D.13 SDR Error Count Format

Table D-13 SDR Error Count File Format

Name	Units	Fmt	Description/Computation
NumRecZeroFill	counts	I6	Number Records Zero Filled
NumNotinFTRK	counts	I6	Number Records not in Fine Track
NumBackScatterErr	counts	I6	Number Records Flagged Backsctter Error
NumRcvrTempErr	counts	I6	Number Records Flagged Receiver Temperature Error
NumVATTEstErr	counts	I6	Number Records Flagged VATT Estimate Error
NumNoSmoothVATT	counts	I6	Number Records Flagged No Smoothed VATT
NumRateErr	counts	I6	Number Records Flagged Rate Error
NumSWHBoundsErr	counts	I6	Number Records Flagged SWH Bounds Error
NumAGCBoundsErr	counts	I6	Number Records Flagged AGC Bounds Error
NumHgtBoundsErr	counts	I6	Number Records Flagged Height Bounds Error
NumDFBTempErr	counts	I6	Number Records Flagged DFB Temperature Error
NumRcvr2TempErr	counts	I6	Number Records Flagged Receiver 2 Temperature Error
NumRcvr1TempErr	counts	I6	Number Records Flagged Receiver 1 Temperature Error
NumTrs2TempErr	counts	I6	Number Records Flagged Trs 2 Temperature Error
NumTrs1TempErr	counts	I6	Number Records Flagged Trs 1 Temperature Error
NumOffNadirErr	counts	I6	Number Records Flagged Off nadir Error
NumSWHSTDErr	counts	I6	Number Records Flagged SWH STD Error
NumAGCSTDErr	counts	I6	Number Records Flagged AGC STD Error
NumHeightSTDErr	counts	I6	Number Records Flagged Height STD Error
NumFiveFramesMiss	counts	I6	Number Records Flagged more than Five Frames Missing
TotDelete	counts	I6	Total Number Records Deleted
TotUsed	counts	I6	Total Number Records Used
TotFrames	counts	I6	Total Number Records Available
Note: TAB delimited file			

D.14 RA Calibration with Waveform Database Format

Table D-14 RA Calibration with Waveform Database Format

Name	Units	Fmt	Description/Computation
Seconds	sec	F16.4	UTC Time in seconds from J2000
ATB	date	A20	UTC Date and Time
Latitude	id	F13.6	Latitude (-90 to +90)
Stat	id	A4	Min/Max/RMS/Mean/Full
TestID	txt	A20	RA Configuration (Ex: RA 1 SSPA 1)
Segment	txt	A20	Segment ID (filename)
RecCount	counts	F5.0	Number of records used in computing average
TrkFmt	txt	A5	Track Format (CAL1,CAL2)
Height	meters	F18.8	Height
AGC	dB	F13.6	AGC
SWH	m	F13.6	SWH
VAT	ratio	F13.6	VATT
AGCGateAmp	counts	F13.6	Scaled counts*(Waveform Scaling Factor+1)
LEBinCnt	counts	F13.6	Scaled counts*(Waveform Scaling Factor+1)
EarlyGateAmp	counts	F13.6	Scaled counts*(Waveform Scaling Factor+1)
LateGateAmp	counts	F13.6	Scaled counts*(Waveform Scaling Factor+1)
MidGateAmp	counts	F13.6	Scaled counts*(Waveform Scaling Factor+1)
AttGateAmp	counts	F13.6	Scaled counts*(Waveform Scaling Factor+1)
NoiseGateAmp	counts	F13.6	Scaled counts*(Waveform Scaling Factor+1)
SignalNoise	counts	F13.6	Scaled counts*(Waveform Scaling Factor+1)
SubMode	counts	F13.6	Scaled counts*(Waveform Scaling Factor+1)
GateIndex	counts	F13.6	Scaled counts*(Waveform Scaling Factor+1)
Gate01	counts	F10.2	Value for Gate 1
Gate02	counts	F10.2	Value for Gate 2
.	.	.	.
.	.	.	.
.	.	.	.
Gate63	counts	F10.2	Value for Gate 63
Gate64	counts	F10.2	Value for Gate 64
Note: SPACE delimited file			

D.15 NGDR Cycle Summary Database Format

Table D-15 NGDR Cycle Summary Database Format

	Field	Units	Format	Description
	TIMESEC	sec	F16.3	Beginning 2000 Epoch of Cycle
	ATB	date	A24	Beginning UTC Time of Cycle
	CYCLE	#	I3	Cycle Number
	STARTDATE	#	A7	Start Date of Cycle (YYYYDOY)
	STOPDATE	#	A7	Stop Date of Cycle (YYYYDOY)
	RECCOUNT	counts	F12.1	Number Frames Used in Cycle after Filter
	SSHCORMEAN	m	F14.4	Cycle Average (SSHC = SSHU-Environment Corrections)
	SSHUNCORMEAN	m	F14.4	Cycle Average (SSHU=SatAtt-StRng+NetHgtCorr)
	SSHUSTDMEAN	m	F14.4	Cycle Average (Standard Deviation from Fit applied to SSHU Values)
	WINDSPEEDMEAN	cm/sec	F14.4	Cycle Average (Wind Speed from Sigma0)
	SWHMEAN	m	F14.4	Cycle Average (Significant Waveheight)
	SWHSTDMEAN	m	F14.4	Cycle Average (Standard Deviation from Fit applied to SWH Values)
	SIGMA0MEAN	dB	F14.4	Cycle Average (Backscatter Coefficient)
	AGCMEAN	dB	F14.4	Cycle Average (Automatic Gain Control)
	AGCSTDMEAN	dB	F14.4	Cycle Average (Standard Deviation from Fit applied to AGC Values)
	DRYTOPOMEAN	m	F14.4	Cycle Average (Dry Troposphere)
	WETTROPOMEAN	m	F14.4	Cycle Average (Wet Troposphere)
	NETAGCCORMEAN	dB	F14.4	Cycle Average (Net AGC Correction)
	ATTITUDEMEAN	deg	F14.4	Cycle Average (Attitude (Off-Nadir Angle))
	IONOMEAN	m	F14.4	Cycle Average (-1 * Att Rng Corr)
	NETSWHCORMEAN	m	F14.4	Cycle Average (Net SWH Correction)
	INVBAROMEAN	m	F14.4	Cycle Average (Inverse Barometer)
	BTEMP22MEAN	deg K	F14.4	Cycle Average (22 GHz Brightness Temp)
	BTEMP37MEAN	deg K	F14.4	Cycle Average (37 GHz Brightness Temp)
	VATTMEAN	v	F14.4	Cycle Average (Averaged VATT)

Table D-15 NGDR Cycle Summary Database Format (Continued)

	Field	Units	Format	Description
	VATTFITMEAN	v	F14.4	Cycle Average (Fitted VATT)
	RECVRTEMPMEAN	deg C	F14.4	Cycle Average (Receiver Temperature)
	EMBIASMEAN	m	F14.4	Cycle Average (Sea State Bias)
	NETHGTCORMEAN	m	F14.4	Cycle Average (Net Height Correction)
	EARTHTIDEMEAN	m	F14.4	Cycle Average (Solid Earth Tide)
	OCEANTIDEMEAN	m	F14.4	Cycle Average (Ocean Water Tide)
	LOADTIDEMEAN	m	F14.4	Cycle Average (Ocean Load Tide)
	POLETIDEMEAN	m	F14.4	Cycle Average (Pole Tide)
	WATERDEPTHMEAN	m	F14.4	Cycle Average (Water Depth)
	GEOIDMEAN	m	F14.4	Cycle Average (Geoid Height)
	SUBMODEMEAN	m	F14.4	Cycle Average (Submode = ((4*TrkMode(1)) + (2*TypeTrack(1))+ (AGCSource(1))+ (4*TrkMode(2))+ (2*TypeTrack(2))+ (AGCSource(2)))/2)
	TOTFRAMESMEAN	counts	F12.1	Cycle Sum (Frames Available)
	TOTUSEDMEAN	counts	F12.1	Cycle Sum (Frames Used)
	TOTDELETEMEAN	counts	F12.1	Cycle Sum (Frames Deleted)
	TOTTRKFRAMESMEAN	counts	F12.1	Cycle Sum (Frames in TRK)
Note: SPACE delimited file				

D.16 RA Engineering Database Format

Table D-16 RA Engineering Database Format

Name	Units	Fmt	Description/Computation
J2Seconds	sec	F16.3	UTC Time in seconds from J2000
ATB	txt	A20	UTC Data and Time
Stat	txt	A4	Min/Max/RMS/Mean/Full
TestID	txt	A20	RA Configuration (Ex: RA1 SSPA1)
Segment	txt	A20	Segment ID (filename)
RecCount	counts	F5.0	Number of records in computing average
EngTemp01	degC	F9.4	RA 1 Receiver Temp
EngTemp02	degC	F9.4	RA 1 TRS 1 Temp
EngTemp03	degC	F9.4	RA 1 TRS 2 Temp
EngTemp04	degC	F9.4	RA 1 Receiver 1 Temp
EngTemp05	degC	F9.4	RA 1 Receiver 2 Temp
EngTemp06	degC	F9.4	RA 1 DFB Temp
EngTemp07	degC	F9.4	RA 2 Receiver Temp
EngTemp08	degC	F9.4	RA 2 TRS 1 Temp
EngTemp09	degC	F9.4	RA 2 TRS 2 Temp
EngTemp10	degC	F9.4	RA 2 Receiver 1 Temp
EngTemp11	degC	F9.4	RA 2 Receiver 2 Temp
EngTemp12	degC	F9.4	RA 2 DFB Temp
CompRecvrTemp	degC	F9.4	Composite Receiver Temp
CoLatitude	deg	F9.4	Co-Latitude(0-180)
Note: SPACE delimited file			

Appendix E
Software Matrix

Table E-1 Software Matrix

	Execution Process	Software/Oracle Utility	Data Source	Output Products/Files
E.1	proc_gfo_ascra	proc_gfo_ascra.f RASciAvg.f RACALAvg.f RAEngAvg.f RASciDB.f RAWFAvg.f	 ra_data ra_cal_data eng_data ra_data ra_cal_data	See Table 3-1
E.2	gfosciavg (*filename*)	gfosciavg.pro readsciavg.pro	(Any Rate) RA (short) Science Average File (Appendix D.1)	Average Plot (Appendix B.2)
E.3	gfoengavg (*filename*)	gfoengavg.pro readengavg.pro	(Any Rate) RA Engineering Average File (Appendix D.2)	Average Plot (Appendix B.3)
E.4	proc_gfo_gdr	proc_gfo_gdr.f GDRSciAvg.f CreateGDRHdrRec.f GDRStatusEvent.f GDRQualityEvent.f	ngdr	.sciavg/Science Average File (Appendix D.10)
E.5	gfo_ngdrpass (*filename*)	gfo_ngdrpass.pro read_gfongdr.pro	60-second NGDR Science Database File (Appendix D.10)	NGDR Day Plot (Appendix B.9)
E.6	gfogdrhist(*filename*)	gfogdrhist.pro read_gfogdr.pro	60-second NGDR Science Database File (Appendix D.10)	NGDR Cycle Plot (Appendix B.11)
E.7	gfogdravg (*filename*)	gfogdravg.pro readgfogdr.pro	(Any Rate) NGDR Science Average File (Appendix D.10)	NGDR Parameter Plot (Appendix B.10)
E.8	gfoghgtagccor (*filename of cal file*)	gfohgtagccor.pro readscirawf.pro readengavg.pro readosc.pro	 RA Calibation Database File (Appendix D.14) 5-minute Engineering Database File (Appendix D.2) SDR Oscillator Drift Database File (Appendix D.12)	Calibration Trend Plot (Appendix B.15)

Table E-1 Software Matrix (Continued)

	Execution Process	Software/Oracle Utility	Data Source	Output Products/Files
E.9	wffine (*filename*)	wffine.pro readscirawf.pro	(Any Rate) RA (long) Science with Waveforms Average File (Appendix D.3)	RA Waveform (postage stamp) plots (Appendix B.16)
E.10	wfcalgate (*filename*)	wfcalgate.pro readscirawf.pro	RA Calibration Database File (Appendix D.14)	Cal2 Gate Plots (Appendix B.19)
E.11	gfocycletrend (*filename*)	gfocycletrend.pro	NGDR Cycle Summary Database File (Appendix D.15)	Cycle Trend Plots (Appendix B.17)
E.12	proc_gfo_sdr	proc_gfo_sdr.f SDRSciAvg.f SDRStatusEvent.f SDRQualityEvent.f WriteGFOOsc.f	sdr	.sciavg/Science Average File (Appendix D.11)
E.13	gfosdravg (*filename*)	gfosdravg.pro *readgfosdr.pro*	(Any Rate)SDR Science Average File (Appendix D.11)	SDR Science Average Plot (Appendix B.6)
E.14	gfosdrjpg (*filename*)	gfosdrjpg.pro readgfosdr.pro	60-second SDR Science Average File (Appendix D.11)	SDR Parameter Plot file for the web.
E.15	gfoerrwww (*filename*)	gfoerrwww.pro readgfoerr.pro		SDR Error Listing for the web
E.16	sdr7daywww (yyddd)	sdr7daywww		
E.17	sdrtodaywww (yyddd)	sdrtodaywww		
E.18	DumpGFO	DumpGFO.f DumpGDR.f DumpRAEng.f DumpRALong.f DumpRAShort.f DumpSDR.f	*ngdr* eng_data ra_cal_data ra_data sdr	Dump Files for each Appendix D.8 Appendix D.6 Appendix D.4 Appendix D.5 Appendix D.7
E.19	load_gfo_ngdr_db.sh sqlldr gfo@develop/password control=gfo_ngdr_hdr.ctl data=gfoNGDRhdr*yyyydoy*.db See Note 1	SQLLoader	Concatenated header records for day	temp_gfo_ngdr_hdr table
E.20	load_gfo_ngdr_db.sh sqlldr gfo@develop/password control=gfo_ngdr_sci.ctl data=gfoNGDRsci*yyyydoy*.db See Note 2	SQLLoader	Concatenated 60 sec avg science records for day	temp_gfo_ngdr_sci table

Table E-1 Software Matrix (Continued)

	Execution Process	Software/Oracle Utility	Data Source	Output Products/Files
E.21	load_gfo_ngdr_db.sh sqlplus –s gfo@develop/password @ run_load_gfo_ngdr_db.sh See Note 3	SQLPlus	temp_gfo_ngdr_hdr table temp_gfo_ngdr_sci table	gfo_ngdr_hdr table gfo_ngdr_sci table
E.22	load_gfo_ra_cal_db.sh sqlldr gfo@develop/password control=gfo_ra_cal.ctl data=gfoCal*yydoy*.db See Note 4	SQLLoader	Concatenated calibration records for day	temp_gfo_ra_cal table
E.23	load_gfo_ra_cal_db.sh sqlplus –s gfo@develop/password @ run_load_gfo_ra_cal_db.sh See Note 5	SQLPlus	temp_gfo_ra_cal table	gfo_ra_cal table
E.24	load_gfo_ra_eng_db.sh sqlldr gfo@develop/password control=gfo_ra_eng.ctl data=gfoEng*yydoy*.db See Note 4	SQLLoader	Concatenated engineering records for day	temp_gfo_ra_eng table
E.25	load_gfo_ra_eng_db.sh sqlplus –s gfo@develop/password @ run_load_gfo_ra_eng_db.sh See Note 5	SQLPlus	temp_gfo_ra_eng table	gfo_ra_eng table
E.26	load_gfo_sdr_osc_db.sh sqlldr gfo@develop/password control=gfo_sdr_osc.ctl data=gfoOsc*yydoy*.db See Note 4	SQLLoader	Concatenated SDR oscillator drift records for day	temp_gfo_sdr_osc table
E.27	load_gfo_sdr_osc_db.sh sqlplus –s gfo@develop/password @ run_load_gfo_sdr_osc_db.sh See Note 5	SQLPlus	temp_gfo_sdr_osc table	gfo_sdr_osc table
E.28	run_gfo_cal_trend.sh sqlplus –s gfo@develop/password@exec_run_cal_trend.sql	SQLPlus	gfo_ra_cal table	gfoTrend*yrdoy*.cal
E.29	run_gfo_eng_trend.sh sqlplus –s gfo@develop/password@exec_run_eng_trend.sql	SQLPlus	gfo_ra_eng table	gfoTrend*yrdoy*.eng
E.30	run_gfo_osc_trend.sh sqlplus –s gfo@develop/password@exec_run_osc_trend.sql	SQLPlus	gfo_sdr_osc table	gfoTrend*yrdoy*.osc
E.31	run_gfo_gdr_cycle.sh sqlplus –s gfo@develop/password@exec_run_gdr_cycle.sql	SQLPlus	gfo_ngdr_hdr table gfo_ngdr_sci table	gfoSycle###.gdr gfo_ngdr_sum table

Table E-1 Software Matrix (Continued)

	Execution Process	Software/Oracle Utility	Data Source	Output Products/Files
E.32	run_gfo_gdr_trend.sh sqlplus –s gfo@develop/pass-word@exec_run_gdr_trend.sql	SQLPlus	gfo_ngdr_sum table	gfoTrend###.###.gdr

Note 1: Load the header data into temporary header table.

Note 2: Load the science data into temporary science table.

Note 3: Once both header and science data have been loaded into the temporary tables, the data is checked for existing entries and the header and science tables are updated.

Note 4: Load the data into the temporary table.

Note 5: Once the data has been loaded in the temporary table, the data is checked for existing entries and the table is updated.

Appendix F
GFO FTS Documentation

F.1 GFO_FTS_Design.doc

The GFO File Transfer System (FTS) is designed to automate the handling of GFO CALVAL data. It is designed to operate autonomously with user intervention at a minimum.

The FTS is written using UNIX shell (/bin/sh) scripts and is based on several facilities of the UNIX operating system. These facilities include (1) an SMTP mail program, (2) an FTP client/server, and (3) an CRON service. The resulting system was designed and written with emphasis on flexibility and ease of maintenance.

Project/Client Organization

The FTS will initially be configured to handle five "projects". A project consists of one or more "clients" who access the same dataset. A separate UNIX account will be created for each project. These accounts will be set up in a rooted file system with FTP-only access rights. The rooted file system will prevent projects from seeing beyond their own home directory. Passwords will be provided to each project. Password changes will be handled locally to insure good password selection and password change compliance. Initial "projects" will include the three data providers, the CALVAL team, and MacArthur.

There is really no distinction between projects who send data and projects who receive data. Each project has its own /in and /out directories and can either send or receive data within a normal FTP session.

Directory Structure

The directory structure is an integral part of the FTS system. Pathnames used in scripts are referenced relative to the directory structure. Figure F-1 illustrates the directory structure. Table F-1 details the function of each directory.

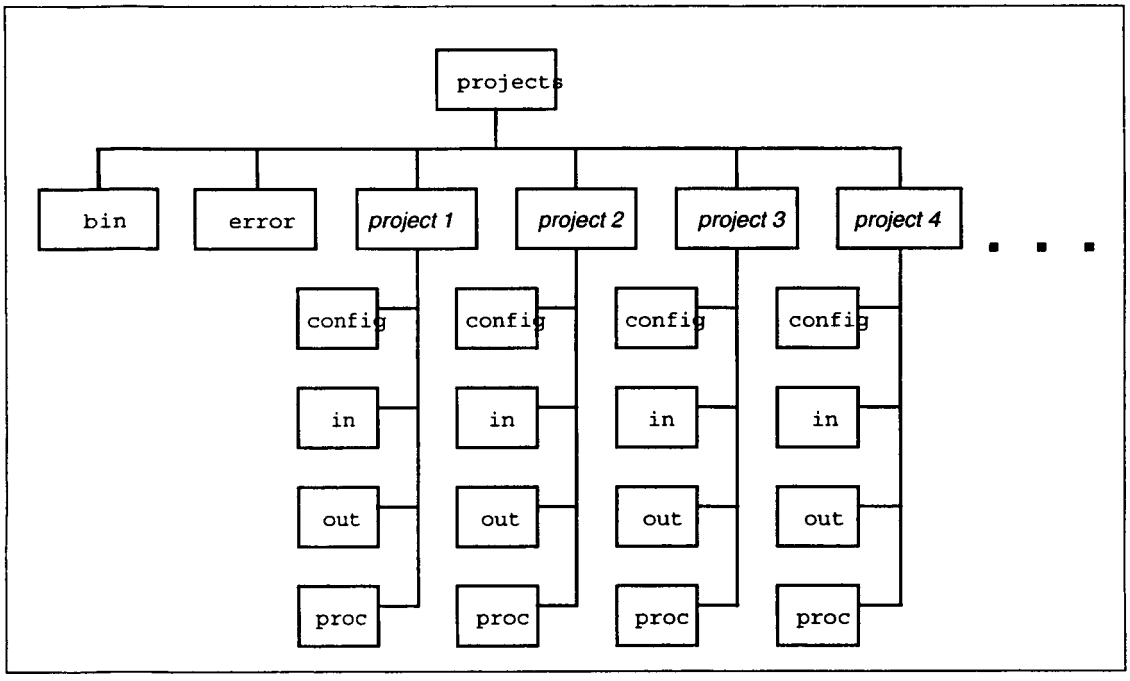

Figure F-1 FTS Directory Structure

Table F-1 FTS Directories

Directory	Function
projects	The main directory that contains all FTS subdirectories
projects/bin	The directory containing executable code (scripts)
projects/error	The directory where unknown files found in the "Inbox" are placed
projects/*project1* projects/*project2* projects/*project3*	Directories containing subdirectories for each project. There is one rooted files system for each project handled by the FTS.
projects/*project1*/config	Directory containing filetype.config files. These config files are user-editable and contain instuctions for what to do with project file sent to the project from WFF.
projects/*project1*/in	"Inbox" Directory where all files to be processed by this system originate. A cron job checks this directory every minute to see if a file has been placed in it. If so, the file is moved to the "proc" directory and processed.
projects/*project1*/out	"Outbox" Directory where a project may retrieve processed files.
projects/*project1*/proc	"Proc" Temporary holding area for files while they are being processed.

Dataflow

Data flows bi-directionally between UNIX scripts, STMP mail, and FTP within the FTS system. Since the directory structure is an integral part of the system, dataflow between directories is also important. All internal dataflow is handled by UNIX scripts and all external dataflow is handled by STMP mail and FTP. Figure F-2 illustrates a simplified diagram of the dataflow of the FTS.

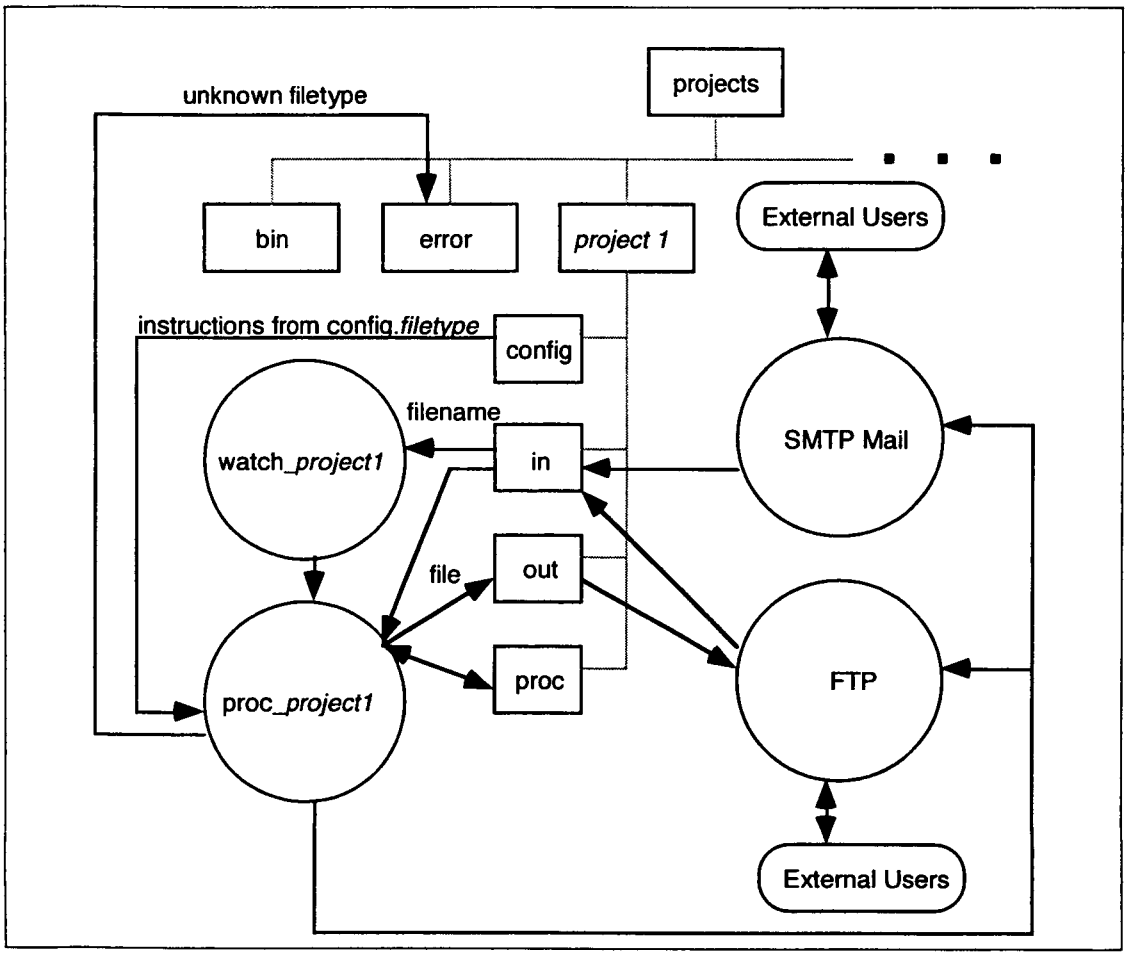

Figure F-2 FTS Dataflow

File Naming Conventions

The FTS works only with all-lowercase filenames. The FTS automatically translates upper-case filenames to an all lower-case equivalent. when it moves a file from the "Inbox" to the "Proc".

The FTS uses a unique file-naming convention to determine what to do with files that are processed within the system. Each unique type of file used by a project must have a filename beginning with a unique 4-character code designated as "filetype". This

"filetype" code must be assigned to the filetype and project by the FTS operator. Files without an appropriate filetype cause an error condition within the FTS and must be handled manually.

Receiving Files

A client may introduce a file into the system by transferring the file into the "inbox" (projects/project/in) via FTP. Since all FTS users have rooted homes, there is no danger of the client accessing files outside his permitted area.

Criteria for receiving file via FTP:

- Files must be put in incoming directory
- Filenames are case-sensitive and must conform to FTS file naming conventions.

The FTS checks each project inbox once per minute. If it sees a file there, it creates a lock file containing the current size in bytes of the file. The next time it checks the inbox, it compares the current size of the file with the size saved in the lock file. If the two match, the FTS assumes the file has completed transfer and moves the file into the proc directory. If the file sizes do not match, the new size is put into the lock file and the process repeats.

Sending Files

There are two basic ways to send files to external users. A file may be sent to an external user via FTP or it may be put in the "outbox" (projects/project/out) for retrieval by an external user using FTP.

The FTP "send" mechanism is somewhat persistent. It can tell if a file has been transferred correctly and will keep trying for up to three times if the transfer is not successful. After a third unsuccessful try, the FTS will send email to the FTS operator informing him of an error condition.

Scripts

Two scripts comprise the core of the FTS: the project-watching script (watch_inbox), and the project-processing script (proc_project). There should be one proc_project script for each project handled by the FTS.

watch_inbox is executed every minute by the UNIX cron facility. If checks the "inbox" (projects/project/in) directory to see if there are any files in it. If it finds one or more files, it moves each file to the processing directory (projects/project/proc) and executes proc_project. proc_project is executed as a "background" process so that watch_inbox may continue without waiting for the processing to complete.

When executed, proc_project attempts to determine the filetype by breaking off the first four characters of the filename. It then looks to see if there is any internal processing to be performed on that filetype. If so, the processing is performed. Unless the script is instructed to exit after internal processing, the script checks for the existence of a filetype.config file. If this file does not exist, an error message is sent to the operator stating that a file of unknown type has been placed in the project's "inbox" and

the file is moved to the error (projects/error) directory. Otherwise, if a filetype.config file exists, it is read and parsed to perform the functions specified within the config file. At the end of processing, the original file is deleted from the processing (projects/project/proc) directory.

.config files

.config files are a simple method of allowing the FTS custodian to customize what happens to files that are received by the. There is one *filetype.config* file for each project *filetype* processed by the FTS. These configuration files are user-editable text files containing a keyword and arguments. A configuration may contain many different directives and each directive will be parsed and followed. The README file with details regarding .config file fomat and directives follows.

```
# README file for modifying .config files
# (Lines beginning with '#' are considered comments)
# ---------------------------------
# Configuration file are named config.FILETYPE. This file controls what
# action(s) will be taken when a file of type FILETYPE is processed.
# The format of this file is one action per line with tab or
# or space-delimited arguments.
#
# Supported actions and required arguments follow.
#
# Syntax of this file may be checked using the 'checkconfig' script.
#
# ACTION TYPE 1 - MOVE
# -------------------
# This action moves the 'radr' file to another directory (usually outbox).
# Optional UPPER or LOWER keyword changes case of filename appropriately.
# The TRIM keyword (alone or with UPPER & LOWER) removes trailing extensions
# (anything after a period (.)) in the filename.
#
#  MOVE directory {UPPER | LOWER | TRIM | UPPER-TRIM | LOWER-TRIM}
#
```

```
#    eg: MOVE /disk2/radarsat/out
#
# ACTION TYPE 2 - FTP
# ------------------
# This action FTPs the 'radr' file to a remote host and places it in the
# specified directory on that host. User and password are necessary to
# gain access to remote system. Anonymous is allowed if remote host
# accepts anonymous FTP. Optional UPPER or LOWER keyword changes case
# of filename appropriately. The TRIM keyword (alone or with UPPER & LOWER)
# removes trailing extensions (anything after a period (.)) in the filename.
#
#  FTP host user pass remote-file | dir {UPPER | LOWER | TRIM | UPPER-
TRIM | LOWER-TRIM}
#
#    eg: FTP radarsat.nasa.gov anonymous radar@woss2.wff.nasa.gov /pub
#
# ACTION TYPE 3 - NOTIFY
# ----------------------
#
# This action notifies the email user upon receipt of a file. The
# project-filename is the subject of the email message and the body
# of the message contains a line giving the project, filename, and
# time received.
#
#  NOTIFY SMTP_address
#
#    eg: NOTIFY joe@radarsat.nasa.gov
#
# ACTION TYPE 4 - MAIL
# --------------------
# This action mails the 'radr' file to an SMTP address.
```

Optional UPPER or LOWER keyword changes case of filename appropriately.
The TRIM keyword (alone or with UPPER & LOWER) removes trailing extensions
(anything after a period (.)) in the filename.
#
MAIL SMTP_address {UPPER | LOWER | TRIM | UPPER-TRIM | LOWER-TRIM}
#
eg: MAIL joe@radarsat.nasa.gov
#
ACTION TYPE 5 - PROCESS

This action processes the 'radr' file with the script named in the
second argument. The script named must be executable and found in the
/wotrsfts/bin directory.
#
PROCESS script_file
#
eg: PROCESS my_script_file.sh
#
#==========
End of README.config

A sample .config file follows. This config file directs the processor to FTP an input file to osb1, notify jlee that the file has been received, and then move the file to another area.

FTP "osb1.wff.nasa.gov" "anonymous" "jlee@osb.wff.nasa.gov" "/pub"
NOTIFY "jlee@osb.wff.nasa.gov"
MOVE "/disk2/jlee"

Logging

All actions performed by this system are logged project-by-project in the log directory. A sample logfile is shown below:

09/09/98 13:30:03 watch_navo Move /navo/in/ra_data98174_05_56_03.dat to /navo/proc/ra_data98174_05_56_03.dat.

09/09/98 13:30:04 watch_navo Running proc_navo.

09/09/98 13:30:04 proc_navo Processing ra_data98174_05_56_03.dat, File Type: ra_d.

09/09/98 13:30:14 proc_config.sh Config: Move ra_data98174_05_56_03.dat to /wff/in.

09/09/98 13:30:24 proc_config.sh Config: Move ra_data98174_05_56_03.dat to /calval/in.

09/09/98 13:30:24 watch_navo Move /navo/in/ra_data98174_09_11_14.dat to /navo/proc/ra_data98174_09_11_14.dat.

09/09/98 13:30:24 watch_navo Running proc_navo.

09/09/98 13:30:24 proc_navo Processing ra_data98174_09_11_14.dat, FileType: ra_d.

09/09/98 13:30:28 watch_navo Move /navo/in/ra_data98174_19_47_54.dat to /navo/proc/ra_data98174_19_47_54.dat.

09/09/98 13:30:28 watch_navo Running proc_navo.

09/09/98 13:30:28 proc_navo Processing ra_data98174_19_47_54.dat, FileType: ra_d.

09/09/98 13:30:43 proc_config.sh Config: Move ra_data98174_19_47_54.dat to /wff/in.

09/09/98 13:30:49 watch_navo Move /navo/in/ra_data98174_23_05_38.dat to /navo/proc/ra_data98174_23_05_38.dat.

09/09/98 13:30:50 watch_navo Running proc_navo.

09/09/98 13:30:50 proc_navo Processing ra_data98174_23_05_38.dat, FileType: ra_d.

09/09/98 13:31:00 proc_config.sh Config: Move ra_data98174_19_47_54.dat to /calval/in.

09/09/98 13:31:18 proc_config.sh Config: Move ra_data98174_09_11_14.dat to /wff/in.

09/09/98 13:31:26 proc_config.sh Config: Move ra_data98174_23_05_38.dat to /wff/in.

09/09/98 13:32:04 proc_config.sh Config: Move ra_data98174_23_05_38.dat to /calval/in.

09/09/98 13:32:27 proc_config.sh Config: Move ra_data98174_09_11_14.dat to /calval/in.

F.2 GFO_FTS_Users_Guide.doc

The GFO File Transfer System (FTS) is designed to automate the handling of GFO CALVAL data. It is designed to operate autonomously with user intervention at a minimum.

Site Information

Hostname calval.wff.nasa.gov

Contacts

David Hancock
NASA GSFC/WFF
hancock@osb.wff.nasa.gov
757-824-1238

Dennis Lockwood
Raytheon Corporation; NASA GSFC/WFF
lockwood@osb.wff.nasa.gov
757-824-2252

Jeff Lee
Raytheon Corporation; NASA GSFC/WFF
jlee@osb.wff.nasa.gov
757-824-1853

Capabilities

- The FTS will "push" data to clients via an automated FTP.
- The FTS will allow clients to "pull" data via FTP from an online repository.

File Naming Conventions

The FTS works only with all-lowercase filenames. The FTS automatically translates upper-case filenames to an all lower-case equivalent.

The FTS uses a unique file-naming convention to determine what to do with files that are processed within the system. Each unique type of file used by a project must have a filename beginning with a unique 4-character code designated as "filetype". This "filetype" code must be assigned to the filetype by the FTS administrator. Files without an appropriate filetype will cause an error condition within the FTS.

Incoming Files

A client may introduce a file into the system by transferring the file into the "inbox" (/in) via FTP.

Criteria for submitting a file to the FTS via FTP are:

- Files must be put in incoming directory (/in)
- Filenames are case-sensitive and must conform to FTS file naming conventions.

The FTS checks each project inbox once per minute. If it sees a file in the inbox, it creates a lock file containing the current size in bytes of the file. The next time it checks the inbox, it compares the current size of the file with the size saved in the lock file. If the two match, the FTS assumes the file has completed transfer and moves the file. If the file sizes do not match, the new size is put into the lock file and the process repeats. Do not be alarmed if files disappear out of the inbox; the FTS is just doing its job.

Outgoing Files

There are two basic methods by which files may be sent to clients. A file may be automatically "pushed" via FTP or it may be put in the "outbox" (/out) for "pull" retrieval.

The FTP "push" mechanism is somewhat persistent. It can discern whether a file has been transferred correctly and will keep trying for up to three times if the transfer is not successful. After a third unsuccessful try, the FTS will send email to the FTS operator informing him of an error condition.

The FTS will keep a two-week (TBD?) repository of data for the GFO CALVAL project. In order to ease client "pulls", the FTS will, on a daily basis, create a text file listing new files which are entered into the repository. The text file will be named in the format:

newfiles.*YYYY_MM_DD*.txt

Security

- The FTS is built upon rooted FTP directories. Clients will not have access to the remainder of the file system.
- Clients will be allowed FTP-only access to the server. Telnet services will not be permitted.
- Passwords will be provided to each client. Password changes will be handled locally to insure good password selection and password change compliance. Multiple clients may be assigned the same userid and password, if deemed appropriate by the FTS administrators.

Logging

All actions performed by this system are logged in a log directory. The administrator can check this logfile if problems occur. A sample logfile is shown below:

09/09/98 13:30:03 watch_navo Move /navo/in/ra_data98174_05_56_03.dat to /navo/proc/ra_data98174_05_56_03.dat.

09/09/98 13:30:04 watch_navo Running proc_navo.

09/09/98 13:30:04 proc_navo Processing ra_data98174_05_56_03.dat, File Type: ra_d.

09/09/98 13:30:14 proc_config.sh Config: Move ra_data98174_05_56_03.dat to /wff/in.

09/09/98 13:30:24 proc_config.sh Config: Move ra_data98174_05_56_03.dat to /calval/in.

09/09/98 13:30:24 watch_navo Move /navo/in/ra_data98174_09_11_14.dat to /navo/proc/ra_data98174_09_11_14.dat.

09/09/98 13:30:24 watch_navo Running proc_navo.

09/09/98 13:30:24 proc_navo Processing ra_data98174_09_11_14.dat, FileType: ra_d.

09/09/98 13:30:28 watch_navo Move /navo/in/ra_data98174_19_47_54.dat to /navo/proc/ra_data98174_19_47_54.dat.

09/09/98 13:30:28 watch_navo Running proc_navo.

09/09/98 13:30:28 proc_navo Processing ra_data98174_19_47_54.dat, FileType: ra_d.

09/09/98 13:30:43 proc_config.sh Config: Move ra_data98174_19_47_54.dat to /wff/in.

09/09/98 13:30:49 watch_navo Move /navo/in/ra_data98174_23_05_38.dat to /navo/proc/ra_data98174_23_05_38.dat.

09/09/98 13:30:50 watch_navo Running proc_navo.

09/09/98 13:30:50 proc_navo Processing ra_data98174_23_05_38.dat, FileType: ra_d.

09/09/98 13:31:00 proc_config.sh Config: Move ra_data98174_19_47_54.dat to /calval/in.

09/09/98 13:31:18 proc_config.sh Config: Move ra_data98174_09_11_14.dat to /wff/in.

09/09/98 13:31:26 proc_config.sh Config: Move ra_data98174_23_05_38.dat to /wff/in.

09/09/98 13:32:04 proc_config.sh Config: Move ra_data98174_23_05_38.dat to /calval/in.

09/09/98 13:32:27 proc_config.sh Config: Move ra_data98174

Appendix G
Configuration File Examples

G.1 Configuration Files for Incoming NAVO Files

Configuration files are used for the distribution of the various incoming GFO files from NAVO, ADFC, Stennis Space Center, Bay St. Louis, MS. The various types of files received from NAVO are: 1) eng_data: 2) ngdr; 3) oodd; 4) ra_cal_data; 5) ra_data; 6) sdr; 7) wvr_data. These files are distributed to **/calval/in/** and **/wff/in/** directories for further processing and disbursement. The configuration files are located in the **/gen/gfo/dist/navo/config** directory.

G.1.1 Example of config.eng_

```
# config.eng_
#=================================================================
# Handles the incoming GFO Engineering files.
#
MOVE /gen/gfo/dist/wff/in/
MOVE /gen/gfo/dist/calval/in/
NOTIFY lockwood@osb.wff.nasa.gov
NOTIFY anet@osb.wff.nasa.gov
#=END=CONFIG=
```

G.1.2 Example of config.ngdr

```
# config.ngdr
#=================================================================
# Handles the incoming GFO NGDR files.
#
MOVE /gen/gfo/dist/wff/in/
MOVE /gen/gfo/dist/calval/in/
NOTIFY lockwood@osb.wff.nasa.gov
NOTIFY anet@osb.wff.nasa.gov
#=END=CONFIG=
```

G.1.3 Example of config.oodd

```
# config.oodd
#=============================================================
# Handles the incoming GFO OODD files.
#
MOVE /gen/gfo/dist/wff/in/
MOVE /gen/gfo/dist/calval/in/
NOTIFY lockwood@osb.wff.nasa.gov
NOTIFY anet@osb.wff.nasa.gov
#=END=CONFIG=
```

G.1.4 Example of config.ra_c

```
# config.ra_c
#=============================================================
# Handles the incoming GFO RA_CAL_DATA files.
#
MOVE /gen/gfo/dist/wff/in/
MOVE /gen/gfo/dist/calval/in/
NOTIFY lockwood@osb.wff.nasa.gov
NOTIFY anet@osb.wff.nasa.gov
#=END=CONFIG=
```

G.1.5 Example of config.ra_d

```
# config.ra_d
#=============================================================
# Handles the incoming GFO RA DATA files.
#
MOVE /gen/gfo/dist/wff/in/
MOVE /gen/gfo/dist/calval/in/
NOTIFY lockwood@osb.wff.nasa.gov
NOTIFY anet@osb.wff.nasa.gov
#=END=CONFIG=
```

G.1.6 Example of config.sdr0

```
# config.sdr0
#=============================================================
# Handles the incoming GFO SDR files.
#
MOVE /gen/gfo/dist/wff/in/
MOVE /gen/gfo/dist/calval/in/
NOTIFY lockwood@osb.wff.nasa.gov
NOTIFY anet@osb.wff.nasa.gov
#=END=CONFIG=
```

G.1.7 Example of config.wvr_

```
# config.wvr_
#=============================================================
# Handles the incoming GFO Water Vapor Radiometer files.
#
MOVE /gen/gfo/dist/wff/in/
MOVE /gen/gfo/dist/calval/in/
NOTIFY lockwood@osb.wff.nasa.gov
NOTIFY anet@osb.wff.nasa.gov
#=END=CONFIG=
```

G.2 Configuration Files for Incoming CALVAL Files from NAVO

Configuration files are used for the distribution of the various incoming GFO files from the **/calval/in/** directory. The various types of files are: 1) eng_data: 2) ngdr; 3) oodd; 4) ra_cal_data; 5) ra_data; 6) sdr; 7) wvr_data. These files are distributed to **/calval/out/** for 15 day retention and ftp'd to remote sites for processing. The configuration files are located in the **/gen/gfo/dist/calval/config** directory.

G.2.1 Example of config.eng_

```
# config.eng_
#=============================================================
# Handles the incoming GFO Engineering files.
#
MOVE /gen/gfo/dist/calval/out/
#=END=CONFIG=
```

G.2.2 Example of config.ngdr

```
# config.ngdr
#=================================================================
# Handles the incoming GFO NGDR files.
#
MOVE /gen/gfo/dist/calval/out/
#=END=CONFIG=
```

G.2.3 Example of config.oodd

```
# config.oodd
#=================================================================
# Handles the incoming GFO OODD files.
#
MOVE /gen/gfo/dist/calval/out/
FTP icesat1.gsfc.nasa.gov gfodata alonso-1 data/
#=END=CONFIG=
```

G.2.4 Example of config.ra_c

```
# config.ra_c
#=================================================================
# Handles the incoming GFO RA files.
#
MOVE /gen/gfo/dist/calval/out/
FTP icesat1.gsfc.nasa.gov gfodata alonso-1 data/
#=END=CONFIG=
```

G.2.5 Example of config.ra_d

```
# config.ra_d
#=================================================================
# Handles the incoming GFO RA files.
#
MOVE /gen/gfo/dist/calval/out/
#=END=CONFIG=
```

G.2.6 Example of config.sdr0

```
# config.sdr0
#=================================================================
# Handles the incoming GFO SDR files.
#
MOVE /gen/gfo/dist/calval/out/
FTP icesat1.gsfc.nasa.gov gfodata alonso-1 data/
#=END=CONFIG=
```

G.2.7 Example of config.wvr_

```
# config.wvr_
#=============================================================
# Handles the incoming GFO Water Vapor Radiometer files.
#
MOVE /gen/gfo/dist/calval/out/
#=END=CONFIG=
```

G.3 Configuration Files for Incoming WFF Files from NAVO

Configuration files are used for the distribution of the various incoming GFO files from the **/wff/in/** directory. The various types of files are: 1) eng_data: 2) ngdr; 3) oodd; 4) ra_cal_data; 5) ra_data; 6) sdr; 7) wvr_data. These files are distributed to **/data/unprocessed/** for processing. The configuration files are located in the **/gen/gfo/dist/wff/config** directory.

G.3.1 Example of config.eng_

```
# config.eng_
#=============================================================
# Handles the incoming GFO Engineering files.
#
MOVE /gen/gfo/data/unprocessed/
#PROC process_radata
#=END=CONFIG=
```

G.3.2 Example of config.ngdr

```
# config.ngdr
#=============================================================
# Handles the incoming GFO NGDR files.
#
MOVE /gen/gfo/data/unprocessed/
#=END=CONFIG=
```

G.3.3 Example of config.oodd

```
# config.oodd
#=============================================================
# Handles the incoming GFO OODD files.
#
MOVE /gen/gfo/data/unprocessed/
#=END=CONFIG=
```

G.3.4 Example of config.ra_c

```
# config.ra_c
#=============================================================
# Handles the incoming GFO RA_CAL_DATA files.
#
MOVE /gen/gfo/data/unprocessed/
#PROC process_radata
#=END=CONFIG=
```

G.3.5 Example of config.ra_d

```
# config.ra_d
#=============================================================
# Handles the incoming GFO RA_DATA files.
#
MOVE /gen/gfo/data/unprocessed/
#PROC process_radata
#=END=CONFIG=
```

G.3.6 Example of config.sdr0

```
# config.sdr0
#=============================================================
# Handles the incoming GFO SDR00 (Year 2000) files.
#
MOVE /gen/gfo/data/quicklook/
PROC process_sdr
#=END=CONFIG=
```

G.3.7 Example of config.wvr_

```
# config.wvr_
#=============================================================
# Handles the incoming GFO Water Vapor Radiometer files.
#
MOVE /gen/gfo/data/unprocessed/
#=END=CONFIG=
```

G.4 Configuration Files for Incoming WFF Files from WFF

Configuration files are used for the distribution of the various locally generated GFO files from the **/data/unprocessed/** directory. The various type of files are: 1) gfoE; 2) gfoC; 3) gfoO; 4) gfoN. These files are distributed to **/wff/in/** directory for further disbursement. The configuration files are located in the **/gen/gfo/dist/wff/config** directory.

G.4.1 Example of config.gfoE

```
#config.gfoE
#=====================================================
# Handles loading the ra_eng data to the database
#
PROCESS load_gfo_ra_eng_db.sh
NOTIFY anet@osb.wff.nasa.gov
NOTIFY lockwood@osb.wff.nasa.gov
#=END=CONFIG=
```

G.4.2 Example of config.gfoC

```
#config.gfoC
#=====================================================
# Handles loading the ra_cal data to the database
#
PROCESS load_gfo_ra_cal_db.sh
NOTIFY anet@osb.wff.nasa.gov
NOTIFY lockwood@osb.wff.nasa.gov
#=END=CONFIG=
```

G.4.3 Example of config.gfoO

```
#config.gfoO
#=====================================================
# Handles loading the sdr osc drift data to the database
#
PROCESS load_gfo_sdr_osc_db.sh
NOTIFY anet@osb.wff.nasa.gov
NOTIFY lockwood@osb.wff.nasa.gov
#=END=CONFIG=
```

G.4.4 Example of config.gfoN

```
#config.gfoN
#=================================================
# Handles loading the ngdr data to the database
#
MOVE /gen/gfo/dist/wff/dbtemp
PROCESS load_gfo_ngdr_db.sh
NOTIFY anet@osb.wff.nasa.gov
NOTIFY lockwood@osb.wff.nasa.gov
```

G.5 Configuration Files for WFF Files from the Database

Configuration files are used for requesting and distributing various locally generated database reports. The various file types are: 1) TCal; 2) TEng; 3) TOsc; 4) TGDR; 5) gfoT; 6) CGDR; and 7) gfoS. These files are distributed to /wff/in/ directory for further processing or disbursement. The configuration files are located in the /gen/gfo/dist/wff/config directory.

G.5.1 Example of config.TCal

```
#config.TCal
#=============================================================================
# Runs Oracle procedure to create a current calibration trend report
#
PROCESS run_gfo_cal_trend.sh
NOTIFY anet@osb.wff.nasa.gov
NOTIFY lockwood@osb.wff.nasa.gov
#=END=CONFIG=
```

G.5.2 Example of config.TEng

```
#config.TEng
#=============================================================================
# Runs Oracle procedure to create a current engineering trend report
#
PROCESS run_gfo_eng_trend.sh
NOTIFY anet@osb.wff.nasa.gov
NOTIFY lockwood@osb.wff.nasa.gov
#=END=CONFIG=
```

G.5.3 Example of config.TOsc

```
#config.TOsc
#=============================================================================
# Runs Oracle procedure to create a current SDR oscillator drift trend report
#
```

```
PROCESS run_gfo_osc_trend.sh
NOTIFY anet@osb.wff.nasa.gov
NOTIFY lockwood@osb.wff.nasa.gov
#=END=CONFIG=
```

G.5.4 Example of config.TGDR

```
#config.TGDR
#=============================================================================
# Runs Oracle procedure to create a current ngdr trend report
#
PROCESS run_gfo_gdr_trend.sh
NOTIFY anet@osb.wff.nasa.gov
NOTIFY lockwood@osb.wff.nasa.gov
#=END=CONFIG=
```

G.5.5 Example of config.gfoT

```
#config.gfoT
#=============================================================================
# Delivers the GFO trend reports, gfoTrend.cal,
# gfoTrend.eng, gfoTrend.osc, and
# gfoTrend.gdr to the /gen/gfo/data/trend directory.
#
MOVE /gen/gfo/data/trend
NOTIFY anet@osb.wff.nasa.gov
NOTIFY lockwood@osb.wff.nasa.gov
#=END=CONFIG=
```

G.5.6 Example of config.CGDR

```
#config.CGDR
#=============================================================
# Runs Oracle procedure to create a current ngdr cycle report
#
PROCESS run_gfo_gdr_cycle.sh
NOTIFY anet@osb.wff.nasa.gov
NOTIFY lockwood@osb.wff.nasa.gov
#=END=CONFIG=
```

G.5.7 Example of config.gfoS

```
#config.gfoS
#=============================================================
# Delivers the GFO cycle report, gfoSycle###.gdr to the /gen/gfo/data/trend
#  directory.
#
MOVE /gen/gfo/data/trend/GDRSycleSum
NOTIFY anet@osb.wff.nasa.gov
NOTIFY lockwood@osb.wff.nasa.gov
#=END=CONFIG=
```

Appendix H
Processing Script Examples

H.1 Script for Processing Daily RA Data

autogfodailyra

```
#!/bin/csh
#
# Check for argument
#
if ($#argv > 0) then
  set yyddd = $argv[1]
else
 set jday = `date +%j`
 set jyear = `date +%y`
 if ("$jday" == "001") then
   set jday = 365
   @ jyear--
 else
   @ jday--
 endif
 set yyddd = `(echo $jyear $jday | awk '{printf "%02s%03s",$1,$2}')`
 echo $yyddd
endif
#
# set Defaults
#
set execdir = '/gen/gfo/bin'
set datadir = '/gen/gfo/data/unprocessed'
set radir   = '/gen/gfo/data/ra'
set distdir = '/gen/gfo/dist/wff/in'
#
# Create day of data to unzip
#
set dounzip = `(echo $yyddd | awk '{printf "*ata%5s*.gz",$1}')`
#
# gunzip Data
#
echo 'Running gunzip on Data'
```

```
echo ' '
  gunzip $dounzip
echo 'Finished running gunzip'
echo ' '
#
# do GFO Short Science Averages
#
echo 'Creating runfile. ra_data follow.... '
echo ' '
 if ( -f runfile ) then
     rm -f runfile
 endif
foreach fname (`ls ra_data*.dat`)
  echo '1' >> runfile
  echo $fname
  (echo $fname | awk '{printf "%s",substr($0,1,25)}' ) >> runfile
  echo '' >> runfile
  echo '1' >> runfile
  echo '2' >> runfile
  echo '60' >> runfile
  echo '2' >> runfile
  echo '1' >> runfile
end
echo '0' >> runfile
echo 'runfile created. Running proc_gfo_ascra...'
echo 'Processing Short Frame, 60sec Science Averages'
proc_gfo_ascra < runfile
rm -f runfile
echo ' '
echo 'Processing Short Frame Completed: proc_gfo_ascra : Done.'
echo ' '
echo 'Printing Science *.log Sheets'
echo ' '
lprt topex2 gfo_raS*.log
#
# do GFO Long Science Averages
#
echo 'Creating runfile. ra_cal_data follow...'
echo ' '
if (-f runfile) then
```

```
    rm -f runfile
endif
foreach fname (`ls ra_cal_data*.dat`)
    echo '2' >> runfile
    echo $fname
    (echo $fname | awk '{printf "%s",substr($0,1,29)}' ) >> runfile
    echo '' >> runfile
    echo '1' >> runfile
    echo '2' >> runfile
    echo '60' >> runfile
    echo '2' >> runfile
    echo '1' >> runfile
end
echo '0' >> runfile
echo 'runfile created.  Running proc_gfo_ascra...'
echo 'Processing Long Frame, 5 min Science Averages'
proc_gfo_ascra < runfile
rm -f runfile
echo ' '
echo 'Processing Long Frame Completed: proc_gfo_ascra : Done.'
echo ' '
echo 'Printing Science *.log Sheets'
echo ' '
lprt topex2 gfo_raL*.log
#
# Concatenate Science Average files for plotting
#
set gfosciavgdat = `(echo $yyddd | awk '{printf "gfosciavg%5s.dat",$1}')`
set gfosciavg = `(echo $yyddd | awk '{printf "gfo_ra*%5s*.sciavg",$1}')`
cat $gfosciavg | grep -v Seconds > $gfosciavgdat
#
echo 'Sorting catted Science Average Files'
echo ' '
sort -o $gfosciavgdat -n -k 1 $gfosciavgdat
gfosciavg $gfosciavgdat
#
set gfoEventdat = `(echo $yyddd | awk '{printf "gfoEvent%5s.dat",$1}')`
grep 'S/A Mode' gfo_ra*.event > $gfoEventdat
grep 'BIT' gfo_ra*.event >> $gfoEventdat
grep 'CAL' gfo_ra*.event >> $gfoEventdat
```

```
sort -o $gfoEventdat -n -k 5 $gfoEventdat
lprw topex2 $gfoEventdat
#cat $gfoEventdat | pr | lp -dtopex2
#
# do GFO Engineering Averages
#
echo 'Creating runfile. ra_data follow...'
echo ' '
if (-f runfile) then
    rm -f runfile
endif
foreach fname (`ls eng_data*.dat`)
    echo '3' >> runfile
    echo $fname
    (echo $fname | awk '{printf "%s",substr($0,1,26)}' ) >> runfile
    echo '' >> runfile
    echo '2' >> runfile
    echo '2' >> runfile
    echo '600' >> runfile
    echo '1' >> runfile
end
echo '0' >> runfile
echo 'runfile created.  Running proc_gfo_ascra...'
echo 'Processing Engineering, 60sec Engineering Averages'
proc_gfo_ascra < runfile
#
# Concatenate Engineering average files for Eng Database
#
set gfoengdb = `(echo $yyddd | awk '{printf "gfoEng%5s.db",$1}')`
set gfoengavg = `(echo $yyddd | awk '{printf "gfo_raE*%5s*.engavg",$1}')`
cat $gfoengavg > $gfoengdb
cp $gfoengdb $distdir/
rm -f runfile
echo ' '
echo 'Processing Engineering Completed: proc_gfo_ascra : Done.'
echo ' '
#
# do GFO Calibration Averages
#
echo 'Creating runfile. ra_cal_data follow...'
```

```
echo ' '
if (-f runfile) then
    rm -f runfile
endif
foreach fname (`ls ra_cal_data*.dat`)
    echo '2' >> runfile
    echo $fname
    (echo $fname | awk '{printf "%s",substr($0,1,29)}' ) >> runfile
    echo '' >> runfile
    echo '3' >> runfile
    echo '2' >> runfile
    echo '600' >> runfile
    echo '1' >> runfile
end
echo '0' >> runfile
echo 'runfile created. Running proc_gfo_ascra...'
echo 'Processing Long Frame, 10sec Cal1 & 1min Cal2 Averages'
proc_gfo_ascra < runfile
#
# Concatenate Calibration average files for Cal Database
#
set gfocaldb = `(echo $yyddd | awk '{printf "gfoCal%5s.db",$1}')`
set gfocalavg = `(echo $yyddd | awk '{printf "gfo_raL*%5s*.wfcavg",$1}')`
cat $gfocalavg > $gfocaldb
cp $gfocaldb $distdir/
rm -f runfile
echo ' '
echo 'Processing Long Frame Completed: proc_gfo_ascra : Done.'
echo ' '
#
# gzip Data
#
set dozip = `(echo $yyddd | awk '{printf "*ata%5s*.dat",$1}')`
echo 'Running gzip on Data'
echo ' '
  gzip $dozip
echo 'Finished running gzip'
echo ' '
#
mv -f $gfosciavgdat $radir/
```

```
mv -f $gfoengdb $radir/
mv -f $gfocaldb $radir/
mv -f gfo_ra*.log $radir/
#rm gfo_ra*.log
rm gfo_ra*.event*
rm gfo_ra*.sciavg*
rm gfoEvent*.dat*
#
echo 'autogfodailyra : Done '
#
```

H.2 Script for Processing Daily SDR Data

autogfodailysdr

```
#!/bin/csh
#
# Check for argument
#
if ($#argv > 0) then
  set yyddd = $argv[1]
else
 set jday = `date +%j`
 set jyear = `date +%y`
 if ("$jday" == "001") then
   set jday = 365
   @ jyear--
 else
   @ jday--
 endif
 set yyddd = `(echo $jyear $jday | awk '{printf "%02s%03s",$1,$2}')`
 echo $yyddd
endif
#
# set Defaults
#
set execdir = '/gen/gfo/bin'
set datadir = '/gen/gfo/data/unprocessed'
set sdrdir  = '/gen/gfo/data/sdr'
set wwwdir  = '/gen/gfo/data/www'
set distdir = '/gen/gfo/dist/wff/in'
#
# Change to processing directory
#
cd $datadir
#
# Create day of data to unzip
#
set dounzip = `(echo $yyddd | awk '{printf "sdr%5s*dat.gz",$1}')`
#
# gunzip Data
#
```

```
echo 'Running gunzip on Data'
echo ' '
  gunzip $dounzip
echo 'Finished running gunzip'
echo ' '
#
# do GFO SDR Science Averages for dbase
#
echo 'Creating runfile. sdr follow...'
echo ' '
if (-f runfile) then
    rm -f runfile
endif
foreach fname (`ls sdr*.dat`)
    echo '1' >> runfile
    echo $fname >> runfile
#    (echo $fname | awk '{printf "%s",substr($0,1,29)}' ) >> runfile
    echo '-1' >> runfile
end
echo '0' >> runfile
echo 'runfile created.  Running proc_gfo_sdr...'
echo 'Processing SDR Data, 60sec DBase Averages'
$execdir/proc_gfo_sdr < runfile
rm -f runfile
echo ' '
echo 'Processing SDR DBase Completed: proc_gfo_sdr : Done.'
echo ' '
echo 'Printing *.log Sheets'
echo ' '
lprt topex2 gfo_sdr*.log
#
# Concatenate Science Database files for saving
# Concatenate Oscillator Drift files for Osc Database
#
set gfosdrdb = `(echo $yyddd | awk '{printf "gfo_sdr%5s.db",$1}')`
set gfosdrscidb = `(echo $yyddd | awk '{printf "gfo_sdr%5s*.scidb",$1}')`
set gfosdrosc = `(echo $yyddd | awk '{printf "gfo_sdr%5s*.osc",$1}')`
set gfosdroscdb = `(echo $yyddd | awk '{printf "gfoOsc%5s.db",$1}')`
cat $gfosdrscidb > $gfosdrdb
cat $gfosdrosc > $gfosdroscdb
```

```
$execdir/gfosdravg $gfosdrdb
$execdir/gfosdrjpg $gfosdrdb
mv -f *.jpg $wwwdir/
#
set gfosdrerrdb = `(echo $yyddd | awk '{printf "gfo_sdr%5s.errdb",$1}')`
set gfosdrerr = `(echo $yyddd | awk '{printf "gfo_sdr%5s*.err",$1}')`
cat $gfosdrerr > $gfosdrerrdb
$execdir/gfoerrwww $gfosdrerrdb
mv -f gfoerr*.html $wwwdir/
#
# preparing for event printout
#
#set gfoEventdat = `(echo $yyddd | awk '{printf "gfosdrEvent%5s.dat",$1}')`
#grep 'S/A Mode' gfo_sdr*.event > $gfoEventdat
#grep 'BIT' gfo_sdr*.event >> $gfoEventdat
#grep 'CAL' gfo_sdr*.event >> $gfoEventdat
#sort -o $gfoEventdat -n -k 5 $gfoEventdat
#lprw topex2 $gfoEventdat
#
#  Run HTML scripts
#
$execdir/sdr7daywww "$yyddd"
$execdir/sdrtodaywww "$yyddd"
#
# gzip Data
#
set dozip = `(echo $yyddd | awk '{printf "sdr%5s*.dat",$1}')`
echo 'Running gzip on Data'
echo ' '
  gzip $dozip
echo 'Finished running gzip'
echo ' '
#
cp $gfosdroscdb $distdir/
mv -f $gfosdrdb $sdrdir/
mv -f $gfosdrerrdb $sdrdir/
mv -f $gfosdroscdb $sdrdir/
#
rm gfo_sdr*.log
rm gfo_sdr*.err
```

```
rm gfo_sdr*.event
rm gfo_sdr*.scidb
#
echo 'autogfodailysdr : Done '
#
```

H.3 Script for Processing Daily NGDR Data

autogfodailyngdr

```
#!/bin/csh
#
# Check for argument
#
if ($#argv > 0) then
  set yyddd = $argv[1]
else
 set jday = `date +%j`
 set jyear = `date +%y`
 if ("$jday" == "001") then
   set jday = 365
   @ jyear--
 else
   @ jday--
 endif
 set yyddd = `(echo $jyear $jday | awk '{printf "%02s%03s",$1,$2}')`
 echo $yyddd
endif
#
# set Defaults
#
set execdir = '/gen/gfo/bin'
set datadir = '/gen/gfo/data/unprocessed'
set ngdrdir = '/gen/gfo/data/ngdr'
set distdir = '/gen/gfo/dist/wff/in'
#
# Change to processing directory
#
cd $datadir
#
# Create day of data to unzip
#
set dounzip = `(echo $yyddd | awk '{printf "ngdr_gfoo_20%5s*.gz",$1}')`
#
# gunzip Data
#
echo 'Running gunzip on Data'
```

```
echo ' '
  gunzip $dounzip
echo 'Finished running gunzip'
echo ' '
#
# do GFO NGDR Science Averages for dbase
#
set unzipped = `(echo $yyddd | awk '{printf "ngdr_gfoo_20%5s*",$1}')`
echo 'Creating runfile. ngdr follow...'
echo ' '
if (-f runfile) then
    rm -f runfile
endif
foreach fname (`ls $unzipped`)
    echo '1' >> runfile
    echo $fname >> runfile
    echo '-1' >> runfile
end
echo '0' >> runfile
echo 'runfile created.  Running proc_gfo_gdr...'
echo 'Processing NGDR Data, 60sec DBase Averages'
proc_gfo_gdr < runfile
rm -f runfile
echo ' '
echo 'Processing NGDR DBase Completed: proc_gfo_gdr : Done.'
echo ' '
echo 'Printing *.log Sheets'
echo ' '
lprt topex2 gfo_ngdr*.log
#
# Concatenate NGDR Science Averages for NGDR Database
#
set gfongdrhdrdb = `(echo $yyddd | awk '{printf "gfoNGDRhdr20%5s.db",$1}')`
set gfongdrscidb = `(echo $yyddd | awk '{printf "gfoNGDRsci20%5s.db",$1}')`
set gfongdrhdravg = `(echo $yyddd | awk '{printf "gfo_ngdr20%5s*.hdrdb",$1}')`
set gfongdrsciavg = `(echo $yyddd | awk '{printf "gfo_ngdr20%5s*.scidb",$1}')`
cat $gfongdrhdravg >> $gfongdrhdrdb
cat $gfongdrsciavg >> $gfongdrscidb
gfo_ngdrpass $gfongdrscidb
gfogdravg $gfongdrscidb
```

```
#
# gzip Data
#
set dozip = `(echo $yyddd | awk '{printf "ngdr_gfoo_20%5s*",$1}')`
echo 'Running gzip on Data'
echo ' '
  gzip $dozip
echo 'Finished running gzip'
echo ' '
#
cp $gfongdrhdrdb $distdir/
cp $gfongdrscidb $distdir/
mv -f $gfongdrhdrdb $ngdrdir/
mv -f $gfongdrscidb $ngdrdir/
mv -f gfo_ngdr*.hdrdb $ngdrdir/
mv -f gfo_ngdr*.scidb $ngdrdir/
rm gfo_ngdr*.log
rm gfo_ngdr*.event
#
echo 'autogfodailyngdr : Done '
#
```

H.4 Script for Quick Look SDR Process

autoquicklooksdr

```
#!/bin/csh
#
# Check for argument
#
if ($#argv > 0) then
  set filename = $argv[1]
else
  echo 'Enter filename in format sdr00000_00_00_00_00000.dat.gz'
endif
#
# set Defaults
#
set execdir = '/gen/gfo/bin'
set datadir = '/gen/gfo/data/quicklook'
set sdrdir  = '/gen/gfo/data/sdr'
#
# Create day of data to unzip
#
#set dounzip = `(echo $yyddd | awk '{printf "sdr%5s*.gz",$1}')`
#
# gunzip Data
#
echo 'Running gunzip on Data'
echo ' '
  gunzip $filename
echo 'Finished running gunzip'
echo ' '
#
# do GFO SDR Science Averages for dbase
#
echo 'Creating runfile. sdr follow...'
echo ' '
if (-f runfile) then
    rm -f runfile
endif
#
set corename=`basename $filename ".gz"`
```

```
set UTC=`basename $corename ".dat"`
      echo '1' >> runfile
      echo $corename >> runfile
      echo '30' >> runfile
      echo '0' >> runfile
      echo '1' >> runfile
      echo '0' >> runfile
echo 'runfile created.  Running proc_gfo_sdr...'
echo 'Processing SDR Data, 30 sec Averages'
proc_gfo_sdr < runfile
rm -f runfile
echo ' '
echo 'Processing SDR Averages Completed: proc_gfo_sdr : Done.'
echo ' '
echo 'Printing *.log Sheet'
echo ' '
set gfosdrlog = `(echo $UTC | awk '{printf "gfo_%s.log",$1}')`
#lprt topex3 $gfosdrlog
#
set gfosdrsciavg = `(echo $UTC | awk '{printf "gfo_%s.sciavg",$1}')`
gfosdrquicklook $gfosdrsciavg
#
# gzip Data
#
echo 'Running gzip on Data'
echo ' '
  gzip $corename
echo 'Finished running gzip'
echo ' '
#
echo 'autoquicklooksdr : Done '
#
```

H.5 Script for Per-Cycle GFO-NCEP Co-locate Process

do_coloc_gfo_ncep_cycle

```
#!/usr/bin/csh

set DIR_GFO_INPUT = /raid/gfo/ngdr03106-03122.C51
set DIR_NCEP_OUT = /gen/ncep/COLOC_GFO_NCEP/coloc.C51

gunzip ${DIR_GFO_INPUT}/gfo_ngdr*.dump.gz

foreach DATA_FILE (${DIR_GFO_INPUT}/gfo_ngdr*.dump)

  if (-f runfile) then
    rm -f runfile
  endif
  if (-f runfile2) then
    rm -f runfile2
  endif
  if (-f runfile3) then
    rm -f runfile3
  endif
  if (-f runfile4) then
    rm -f runfile4
  endif
  if (-f runfile5) then
    rm -f runfile5
  endif
  if (-f runfile6) then
    rm -f runfile6
  endif
  if (-f runfile7) then
    rm -f runfile7
  endif

  if (-f OUT_do_coloc_gfo_ncep) then
    rm -f OUT_do_coloc_gfo_ncep
  endif

  ln -s $DATA_FILE .
  ls gfo_ngdr*.dump > runfile
```

```
  echo "working on ${DATA_FILE} ..."

# determination of the date for the ncep file from the name of the gfo ngdr

  echo -n "20" > runfile2
  (echo $DATA_FILE | awk '{printf "%s",substr($0,39,2)}' ) >> runfile2
  echo -n "-" >> runfile2
  (echo $DATA_FILE | awk '{printf "%s",substr($0,41,3)}' ) >> runfile2
  echo -n "T12:00:00.000000" >> runfile2
#   more runfile2
  decode_atbcal < runfile2

# get the ncep files for these dates

  echo -n "ln -s /gen/ncep/Data" > runfile3
  (more date_for_ncep_1 | awk '{printf "%s",substr($0,1,4)}' ) >> runfile3
  echo -n "/U." >> runfile3
  (more date_for_ncep_1 | awk '{printf "%s",substr($0,1,6)}' )>> runfile3
  echo -n ".* ." >> runfile3
  echo " " >> runfile3
  echo -n "ln -s /gen/ncep/Data" >> runfile3
  (more date_for_ncep_2 | awk '{printf "%s",substr($0,1,4)}' ) >> runfile3
  echo -n "/U." >> runfile3
  (more date_for_ncep_2 | awk '{printf "%s",substr($0,1,6)}' ) >> runfile3
  echo -n ".00 ." >> runfile3
  echo " " >> runfile3
  echo -n "ln -s /gen/ncep/Data" >> runfile3
  (more date_for_ncep_1 | awk '{printf "%s",substr($0,1,4)}' ) >> runfile3
  echo -n "/V." >> runfile3
  (more date_for_ncep_1 | awk '{printf "%s",substr($0,1,6)}' ) >> runfile3
  echo -n ".* ." >> runfile3
  echo " " >> runfile3
  echo -n "ln -s /gen/ncep/Data" >> runfile3
  (more date_for_ncep_2 | awk '{printf "%s",substr($0,1,4)}' ) >> runfile3
  echo -n "/V." >> runfile3
  (more date_for_ncep_2 | awk '{printf "%s",substr($0,1,6)}' ) >> runfile3
  echo -n ".00 ." >> runfile3

#   more runfile3
  chmod a+x runfile3
```

```
  runfile3

# give these ncep file names to the colocation code

  ls U.* >> runfile
  ls V.* >> runfile

#   more runfile

# test if there are all the ncep files we need

  if (-f test_ok) then
    rm -f test_ok
  endif

  if (-f test_date1) then
    rm -f test_date1
  endif
  if (-f test_date2) then
    rm -f test_date2
  endif

  echo -n "cp U." > runfile5
  (more date_for_ncep_1 | awk '{printf "%s",substr($0,1,6)}' ) >> runfile5
  echo ".00 test_date1" >> runfile5
#   more runfile5
  chmod a+x runfile5
  runfile5

  echo -n "cp U." > runfile6
  (more date_for_ncep_2 | awk '{printf "%s",substr($0,1,6)}' ) >> runfile6
  echo ".00 test_date2" >> runfile6
#   more runfile6
  chmod a+x runfile6
  runfile6

#   more test_date1
#   more test_date2

# pour savoir si un fichier est vide (-z en csh)
```

```
  if (-f test_date1 && -f test_date2) then
   echo ok > test_ok
  endif

#   more test_ok

  if (-f test_ok) then
     coloc_gfo_ncep < runfile >> OUT_do_coloc_gfo_ncep

     echo -n "mv OUT_do_coloc_gfo_ncep ${DIR_NCEP_OUT}/gfo_ncep_" > runfile4
     (echo $DATA_FILE | awk '{printf "%s",substr($0,39,2)}' ) >> runfile4
     echo -n "_" >> runfile4
     (echo $DATA_FILE | awk '{printf "%s",substr($0,41,15)}' ) >> runfile4
     echo -n ".dat" >> runfile4

     chmod a+x runfile4
     runfile4
  endif

  rm gfo_ngdr*.dump
  rm U.*
  rm V.*

end

gzip ${DIR_GFO_INPUT}/gfo_ngdr*.dump
```

H.6 Script for Per-Cycle GFO-NCEP Averaging Process

do_avg_and_stat_data

```
#!/usr/bin/csh

set DIR_GFO_INPUT = /raid/gfo/ngdr03106-03122.C51
set DIR_NCEP_INPUT = /gen/ncep/COLOC_GFO_NCEP/coloc.C51
set YEAR = 2003
set CYC = 51

set DIR_OUTPUT = /gen/ncep/COLOC_GFO_NCEP/Data_filter_and_avg

gunzip ${DIR_GFO_INPUT}/gfo_ngdr*.dump.gz

if (-f OUT_do_avg_data) then
  rm -f OUT_do_avg_data
endif
if (-f OUT_do_stat_data) then
  rm -f OUT_do_stat_data
endif

if (-f runfile3) then
  rm -f runfile3
endif
if (-f runfile4) then
  rm -f runfile4
endif

foreach DATA_FILE (${DIR_GFO_INPUT}/gfo_ngdr*.dump)
  if (-f runfile) then
    rm -f runfile
  endif
  if (-f runfile2) then
    rm -f runfile2
  endif

  ln -s $DATA_FILE .
  ls gfo_ngdr*.dump > runfile

  echo -n "ln -s ${DIR_NCEP_INPUT}/gfo_ncep_" > runfile2
```

```
  (echo $DATA_FILE | awk '{printf "%s",substr($0,39,2)}' ) >> runfile2
  echo -n "_" >> runfile2
  (echo $DATA_FILE | awk '{printf "%s",substr($0,41,15)}' ) >> runfile2
  echo -n ".dat ." >> runfile2

  chmod a+x runfile2
  runfile2

  ls gfo_ncep*.dat >> runfile

  echo "working on ${DATA_FILE} ..."

  if (-f test_ok) then
    rm -f test_ok
  endif

  if (-f test_file_gfo_ncep) then
    rm -f test_file_gfo_ncep
  endif

  head gfo_ncep* > test_file_gfo_ncep
  if ( ! -z test_file_gfo_ncep ) then
    echo ok > test_ok
  endif
  more test_ok

  if (-f test_ok) then
    avg_data < runfile >> OUT_do_avg_data
  endif

  rm gfo_ngdr*.dump
  rm gfo_ncep*
end

gzip ${DIR_GFO_INPUT}/gfo_ngdr*.dump

echo "${YEAR}" > runfile4
echo "${CYC}" >> runfile4
do_stat_data < runfile4
```

```
cat OUT_do_stat_data >> gfo_sigma0_summary.dat

echo -n "mv OUT_do_avg_data ${DIR_OUTPUT}/data_avg_" > runfile3
(echo $DATA_FILE | awk '{printf "%s",substr($0,39,2)}' ) >> runfile3
echo -n "_C${CYC}.dat" >> runfile3
chmod a+x runfile3
runfile3
```

H.7 Script for Range Measurement Noise Process

do_highpassfilter_gfo_1min

```
#!/usr/bin/csh

if (-f OUT_do_highpassfilter_gfo_1min) then
    rm -f OUT_do_highpassfilter_gfo_1min
endif

foreach DATA_FILE (gfo_ngdr*.dump)
  if (-f runfile) then
    rm -f runfile
  endif
  echo $DATA_FILE >> runfile
  echo "working on ${DATA_FILE} ..."
  highpassfilter_gfo_1min < runfile >> OUT_do_highpassfilter_gfo_1min
end
#
echo 'Running gzip on dump files'
echo ' '
  gzip gfo_ngdr*.dump
echo 'Finished running gzip'
echo ' '
#
rm -f runfile
echo 'do_highpassfilter_gfo_1min : Done'
#
```

H.8 Script for Quick Look RA Process

autoquicklookra

```
#!/bin/csh
#
# Check for argument
#
if ($#argv > 0) then
  set segment = $argv[1]
else
  echo 'Enter segment in format YYDDD_HH_MM_DD'
endif
#
# set Defaults
#
set execdir = '/gen/gfo/bin'
set datadir = '/gen/gfo/data/unprocessed'
set radir  = '/gen/gfo/data/ra'
#
# Create day of data to unzip
#
set dounzip = `(echo $segment | awk '{printf "*ata%14s*.gz",$1}')`
#
# gunzip Data
#
echo 'Running gunzip on Data'
echo ' '
  gunzip $dounzip
echo 'Finished running gunzip'
echo ' '
#
# clear previous output
#
 rm gfo_ra*.log
 rm gfo_ra*.event
 rm gfo_ra*.sciavg
 rm gforaEvent*.dat
#
echo 'Creating runfile.  ra_data follow.... '
echo ' '
```

```
 if ( -f runfile ) then
     rm -f runfile
 endif
set filename = `(echo $segment | awk '{printf "ra_data%s.dat",$1}')`
foreach fname (`ls ra_data*.dat`)
   echo '1' >> runfile
   echo $fname
   (echo $fname | awk '{printf "%s",substr($0,1,25)}' ) >> runfile
   echo '' >> runfile
   echo '1' >> runfile
   echo '2' >> runfile
   echo '30' >> runfile
   echo '2' >> runfile
   echo '1' >> runfile
end
echo '0' >> runfile
echo 'runfile created.  Running proc_gfo_ascra...'
echo 'Processing Short Frame,  30 sec Science Averages'
proc_gfo_ascra < runfile
rm -f runfile
echo ' '
echo 'Processing Short Frame Completed: proc_gfo_ascra : Done.'
echo ' '
echo 'Printing Science *.log Sheets'
echo ' '
set gforaSlog = `(echo $segment | awk '{printf "gfo_raS_%s.log",$1}')`
#lprt topex3 $gforaSlog
#
# do GFO Long Science Averages
#
echo 'Creating runfile. ra_cal_data follow...'
echo ' '
if (-f runfile) then
     rm -f runfile
endif
set filename = `(echo $segment | awk '{printf "ra_cal_data%s.dat",$1}')`
foreach fname (`ls ra_cal_data*.dat`)
     echo '2' >> runfile
     echo $fname
     (echo $fname | awk '{printf "%s",substr($0,1,29)}' ) >> runfile
```

```
    echo '' >> runfile
    echo '1' >> runfile
    echo '2' >> runfile
    echo '30' >> runfile
    echo '2' >> runfile
    echo '1' >> runfile
end
echo '0' >> runfile
echo 'runfile created.  Running proc_gfo_ascra...'
echo 'Processing Long Frame, 30 sec Science Averages'
proc_gfo_ascra < runfile
rm -f runfile
echo ' '
echo 'Processing Long Frame Completed: proc_gfo_ascra : Done.'
echo ' '
echo 'Printing Science *.log Sheets'
echo ' '
set gforaLlog = `(echo $segment | awk '{printf "gfo_raL_%s.log",$1}')`
#lprt topex3 $gforaLlog
#
# cat gfo_ra*.sciavg
#
set gfosciavgdat = `(echo $segment | awk '{printf "gfosciavg%14s.dat",$1}')`
set gfosciavg = `(echo $segment | awk '{printf "gfo_ra*%14s.sciavg",$1}')`
cat $gfosciavg | grep -v Seconds > $gfosciavgdat
#
echo 'Sorting catted Science File'
echo ' '
sort -o $gfosciavgdat -n -k 1 $gfosciavgdat
gfosciquicklook $gfosciavgdat
#
# preparing for event printout
#
set gforaSevent = `(echo $segment | awk '{printf "gfo_raS_%14s.event",$1}')`
set gforaLevent = `(echo $segment | awk '{printf "gfo_raL_%14s.event",$1}')`
set gfoEventdat = `(echo $segment | awk '{printf "gforaEvent%14s.dat",$1}')`
grep 'S/A Mode' gfo_ra*.event > $gfoEventdat
grep 'BIT' gfo_ra*.event >> $gfoEventdat
grep 'CAL' gfo_ra*.event >> $gfoEventdat
sort -o $gfoEventdat -n -k 5 $gfoEventdat
```

```
#lp -dtopex3 $gfoEventdat
#lprw topex3 $gfoEventdat
#
# gzip Data
#
set dozip = `(echo $segment | awk '{printf "*ata%14s*.dat",$1}')`
echo 'Running gzip on Data'
echo ' '
  gzip $dozip
echo 'Finished running gzip'
echo ' '
#
rm $gforaLlog
rm $gforaSlog
rm $gfoEventdat
rm $gforaSevent
rm $gforaLevent
#rm gfo_ra*.log
#rm gfo_ra*.event
#rm gfo_ra*.sciavg
#rm gforaEvent*.dat
#
mv -f $gfosciavgdat $radir/
#
echo 'autoquicklookra : Done '
#
```

H.9 Script for NGDR Dump Process

gfoallngdrdmp

```
#! /bin/csh
#
# do GFO NGDR Science Dump
#
# gunzip Data
#
echo 'Running gunzip on ngdr files'
echo ' '
  gunzip ngdr_gfo*.gz
echo 'Finished running gunzip'
echo ' '
#
echo 'Creating runfile. ngdr_gfo* follow...'
echo ' '
if (-f runfile) then
    rm -f runfile
endif
foreach fname (`ls ngdr_gfo*`)
    echo '5' >> runfile
    echo $fname >> runfile
    echo '1' >> runfile
end
echo '0' >> runfile
echo 'runfile created.  Running DumpGFO...'
DumpGFO < runfile
rm -f runfile
#
# gzip Files
#
echo 'Running gzip on ngdr files'
echo ' '
  gzip ngdr_gfo*
echo 'Finished running gzip'
echo ' '
#
echo ' '
echo 'gfoallngdrdmp : Done.'
echo ' '
```

Appendix I
Database Scripts

I.1 Scripts for Loading Database Tables

I.1.1 The GFO_NGDR_HDR and GFO_NGDR_SCI Tables

```
#!/bin/sh
#
# load_gfo_ngdr_db.sh [FYTPE] [TEMPDIR] [FILENAME]
#=============================================================
#
# Handle Errors
#
handle_error()
{
 mv $PROCDIR/$FILENAME $ERRORDIR/$FILENAME
 MAILF="/tmp/mail.$$"
 echo "ERROR - No config file exists in $CONFDIR for $FILENAME" > $MAILF
 echo "      Bad file has been moved into $ERRORDIR" >> $MAILF
 $BINMAIL -s "Bad file in $PROCDIR" "$OPERATOR" < $MAILF
 rm -f $MAILF
 echo `date +"%D %T"`"\tproc_config.sh \t" \
  "Error: Cannot process $FILENAME, " \
  " copied to $ERRORDIR."
 exit 1
}
#
#=============================================================
#
# Main Routine
#
# Include globals
#
. /gen/gfo/dist/bin/config.incl
#
# Set arguments
#
FTYPE=$1
TEMPDIR=$2
```

```
FILENAME=$3
#
# Set project
#
PROJECT="wff"
PROCDIR="$ROOTDIR/$PROJECT/proc"
CONFDIR="$ROOTDIR/$PROJECT/config"
OUTDIR="$ROOTDIR/$PROJECT/out"
LOGDIR="$ROOTDIR/log"
LOADLOGFILE="$LOGDIR/$FILENAME.$$.log"
LOGFILE="$LOGDIR/$PROJECT.log"
DBTEMP="$ROOTDIR/$PROJECT/dbtemp"
BADDIR="$ROOTDIR/$PROJECT/bad"
#
if [ "$FTYPE" = "gfoN" ]; then
 BASE=`basename $FILENAME`
 BASE1=`echo $BASE | awk -F. '{printf "%s", $1}'`
 PROJ=`echo $FILENAME | awk '{printf "%s",substr($0,1,3)}'`
 DATYPE=`echo $FILENAME | awk '{printf "%s",substr($0,4,4)}'`
 DBTYPE=`echo $FILENAME | awk '{printf "%s",substr($0,8,3)}'`
 DDATE=`echo $FILENAME | awk '{printf "%s",substr($0,11,7)}'`
 EXT=`echo $FILENAME | awk -F. '{printf "%s",$2}'`
#
 MATE1=$DBTEMP/$PROJ$DATYPE"hdr"$DDATE"."$EXT
 MATE2=$DBTEMP/$PROJ$DATYPE"sci"$DDATE"."$EXT
 case "$DBTYPE" in
  "hdr")
    CONTROL="gfo_ngdr_hdr.ctl"
    ;;
  "sci")
    CONTROL="gfo_ngdr_sci.ctl"
    ;;
  *)
    (handle_error) >> $LOGFILE
    ;;
 esac
#
 echo `date +"%D %T"`"\tproc_$PROJECT   \t" \
     "Loading Oracle with $CONTROL, file $DBTEMP/$FILENAME." >> $LOGFILE
#
```

```
# run SQL Loader to load the data into the database
#
( $ORACLE_HOME/bin/sqlldr gfo@develop/t2gtgg control="$BINDIR/$CONTROL" \
  data="$DBTEMP/$FILENAME" log="$LOADLOGFILE" 2>&1 ) > /dev/null
#
# Print results of database load by grepping the log file.
#
egrep -e "File|Rows|Total" $LOADLOGFILE | pr | lp -dtopex2
#
# Check to see if bad file exists in the bin directory, if so, move to /gen/gfo/dist/wff/bad directo
ry
#
BADFILE="$BASE1.bad"
if [ -f $BINDIR/$BADFILE ]; then
  mv $BINDIR/$BADFILE $BADDIR/$BADFILE
fi
rm $LOADLOGFILE
#
# when all of the gfo files are available, run the stored procedure,
#  load_gfo_ngdr_sci, to update the data in the database
#
 SCRIPTNAME=run_load_gfo_ngdr_db.sh
 SCRIPTERROR=0
 PROCNAME=load_gfo_ngdr_db
 LOCK="$DBTEMP/$PROJ$DATYPE$DDATE.lock"
 PROCESS=1
 REQUIRE="$MATE1 $MATE2"
#
# check to see if all of the gfo files are available, if so process
#
 for TARGET in $REQUIRE
  do
    if [ ! -f $TARGET ]; then
      PROCESS=0
    fi
    if [ -f $TARGET.lock ]; then
      PROCESS=0
    fi
  done
  if [ $PROCESS -eq 1 -a ! -f $LOCK ]; then
```

```
    touch $LOCK
    SCRIPTOUTPUT=$LOGDIR/temp.$$.out
    cat /dev/null > $SCRIPTOUTPUT
#
# run procedure to move files from the temporary tables to the permanent tables
#
    echo `date +"%D %T"`"\tproc_$PROJECT    \t" \
      "Running Procedure $PROCNAME, file $DBTEMP/$FILENAME." >> $LOGFILE
#
    $ORACLE_HOME/bin/sqlplus -s gfo@develop/t2gtgg @$BINDIR/$SCRIPTNAME > $SCRIPTOUTPUT
#
# check success of procedure
#
    SCRIPTERROR=$?
    cat $SCRIPTOUTPUT | grep 'SP2-|ORA-'
    GREPOUTPUT=$?
    if [ $GREPOUTPUT -eq 0 ]; then
      SCRIPTERROR=`expr $SCRIPTERROR + 1`
    fi
    if [ $SCRIPTERROR -ne 0 ]; then
      echo `date +"%D %T"`"\tproc_$PROJECT    \t" \
        "Error Running Procedure $PROCNAME, file $DBTEMP/$FILENAME." >> $LOGFILE
      exit 99
    fi
    rm $DBTEMP/$MATE*
    rm $SCRIPTOUTPUT
    exit 0
  fi
else
 (handle_error) >> $LOGFILE
fi
#
# End of Processing
#
```

I.1.2 The GFO_RA_CAL Table

```
#!/bin/sh
#
# load_gfo_ra_cal_db.sh [FYTPE] [TEMPDIR] [FILENAME]
#=============================================================================
#
```

```
# Handle Errors
#
handle_error()
{
 mv $PROCDIR/$FILENAME $ERRORDIR/$FILENAME
 MAILF="/tmp/mail.$$"
 echo "ERROR - No config file exists in $CONFDIR for $FILENAME" > $MAILF
 echo "      Bad file has been moved into $ERRORDIR" >> $MAILF
 $BINMAIL -s "Bad file in $PROCDIR" "$OPERATOR" < $MAILF
 rm -f $MAILF
 echo `date +"%D %T"`"\tproc_config.sh \t" \
   "Error: Cannot process $FILENAME, " \
   " copied to $ERRORDIR."
 exit 1
}
#
#============================================================================
#
# Main Routine
#
# Include globals
#
. /gen/gfo/dist/bin/config.incl
#
# Set arguments
#
FTYPE=$1
TEMPDIR=$2
FILENAME=$3
#
# Set project
#
ROOTDIR="/gen/gfo/dist"
PROJECT="wff"
PROCDIR="$ROOTDIR/$PROJECT/proc"
CONFDIR="$ROOTDIR/$PROJECT/config"
OUTDIR="$ROOTDIR/$PROJECT/out"
LOGDIR="$ROOTDIR/log"
LOADLOGFILE="$LOGDIR/$FILENAME.$$.log"
LOGFILE="$ROOTDIR/$PROJECT.log"
```

```
BADDIR="$ROOTDIR/$PROJECT/bad"
#
if [ "$FTYPE" = "gfoC" ]; then
 BASE=`basename $FILENAME`
 BASE1=`echo $FILENAME | awk -F. '{printf "%s",$1}'`
 PROJ=`echo $FILENAME | awk -F. '{printf "%s",substr($0,1,3)}'`
 EXT=`echo $FILENAME | awk -F. '{printf "%s",$2}'`
#
 CONTROL="gfo_ra_cal.ctl"
#
 echo `date +"%D %T"`"\tproc_$PROJECT    \t" \
     "Loading Oracle with $CONTROL, file $FILENAME." >> $LOGFILE
#
# run SQL Loader to load the data into the database
#
 ( $ORACLE_HOME/bin/sqlldr gfo@develop/t2gtgg control="$BINDIR/$CONTROL" \
  data="$FILENAME" log="$LOADLOGFILE" 2>&1 ) > /dev/null
#
# Print results of database load by grepping the log file.
#
 egrep -e "File|Rows|Total" $LOADLOGFILE | pr | lp -dtopex2
#
# Check to see if bad file exists in the bin directory, if so, move to /gen/gfo/dist/wff/bad
# directory
#
 BADFILE="$BASE1.bad"
 if [ -f $BINDIR/$BADFILE ] ; then
   mv $BINDIR/$BADFILE $BADDIR/$BADFILE
 fi
 rm $LOADLOGFILE
#
# run the stored procedure,
#  load_gfo_ra_cal_db, to update the data in the database
#
 SCRIPTNAME=run_load_gfo_ra_cal_db.sh
 SCRIPTERROR=0
 PROCNAME=load_gfo_ra_cal_db
 SCRIPTOUTPUT=$LOGDIR/temp.$$.out
 cat /dev/null > $SCRIPTOUTPUT
#
```

```
# run procedure to move files from the temporary tables to the permanent tables
#
 echo `date +"%D %T"`"\tproc_$PROJECT    \t" \
     "Running Procedure $PROCNAME, file $FILENAME." >> $LOGFILE
#
 $ORACLE_HOME/bin/sqlplus -s gfo@develop/t2gtgg @$BINDIR/$SCRIPTNAME > $SCRIPTOUTPUT
#
# check success of procedure
#
 SCRIPTERROR=$?
 cat $SCRIPTOUTPUT | grep 'SP2-|ORA-'
 GREPOUTPUT=$?
 if [ $GREPOUTPUT -eq 0 ]; then
   SCRIPTERROR=`expr $SCRIPTERROR + 1`
 fi
 if [ $SCRIPTERROR -ne 0 ]; then
   echo `date +"%D %T"`"\tproc_$PROJECT    \t" \
      "Error Running Procedure $PROCNAME, file $FILENAME." >> $LOGFILE
   exit 99
 fi
 rm $SCRIPTOUTPUT
 exit 0
else
 (handle_error) >> $LOGFILE
fi
#
# End of Processing
#
```

I.1.3 The GFO_RA_ENG Table

```
#!/bin/sh
#
# load_gfo_ra_eng_db.sh [FYTPE] [TEMPDIR] [FILENAME]
#================================================================================
#
# Handle Errors
#
handle_error()
{
 mv $PROCDIR/$FILENAME $ERRORDIR/$FILENAME
```

```
 MAILF="/tmp/mail.$$"
 echo "ERROR - No config file exists in $CONFDIR for $FILENAME" > $MAILF
 echo "       Bad file has been moved into $ERRORDIR" >> $MAILF
 $BINMAIL -s "Bad file in $PROCDIR" "$OPERATOR" < $MAILF
 rm -f $MAILF
 echo `date +"%D %T"`"\tproc_config.sh \t" \
   "Error: Cannot process $FILENAME, " \
   " copied to $ERRORDIR."
 exit 1
}
#
#=============================================================================
#
# Main Routine
#
# Include globals
#
. /gen/gfo/dist/bin/config.incl
#
# Set arguments
#
FTYPE=$1
TEMPDIR=$2
FILENAME=$3
#
# Set project
#
ROOTDIR="/gen/gfo/dist"
PROJECT="wff"
PROCDIR="$ROOTDIR/$PROJECT/proc"
CONFDIR="$ROOTDIR/$PROJECT/config"
OUTDIR="$ROOTDIR/$PROJECT/out"
LOGDIR="$ROOTDIR/log"
LOADLOGFILE="$LOGDIR/$FILENAME.$$.log"
LOGFILE="$ROOTDIR/$PROJECT.log"
BADDIR="$ROOTDIR/$PROJECT/bad"
#
if [ "$FTYPE" = "gfoE" ]; then
 BASE=`basename $FILENAME`
 BASE1=`echo $FILENAME | awk -F. '{printf "%s",$1}'`
```

```
PROJ=`echo $FILENAME | awk -F. '{printf "%s",substr($0,1,3)}'`
EXT=`echo $FILENAME | awk -F. '{printf "%s",$2}'`
#
CONTROL="gfo_ra_eng.ctl"
#
echo `date +"%D %T"`"\tproc_$PROJECT   \t" \
    "Loading Oracle with $CONTROL, file $FILENAME." >> $LOGFILE
#
# run SQL Loader to load the data into the database
#
( $ORACLE_HOME/bin/sqlldr gfo@develop/t2gtgg control="$BINDIR/$CONTROL" \
  data="$FILENAME" log="$LOADLOGFILE" 2>&1 ) > /dev/null
#
# Print results of database load by grepping the log file.
#
egrep -e "File|Rows|Total" $LOADLOGFILE | pr | lp -dtopex2
#
# Check to see if bad file exists in the bin directory, if so, move to /gen/gfo/dist/wff/bad
# directory
#
BADFILE="$BASE1.bad"
if [ -f $BINDIR/$BADFILE ] ; then
  mv $BINDIR/$BADFILE $BADDIR/$BADFILE
fi
rm $LOADLOGFILE
#
# run the stored procedure,
#  load_gfo_ra_eng_db, to update the data in the database
#
SCRIPTNAME=run_load_gfo_ra_eng_db.sh
SCRIPTERROR=0
PROCNAME=load_gfo_ra_eng_db
SCRIPTOUTPUT=$LOGDIR/temp.$$.out
cat /dev/null > $SCRIPTOUTPUT
#
# run procedure to move files from the temporary tables to the permanent tables
#
echo `date +"%D %T"`"\tproc_$PROJECT   \t" \
   "Running Procedure $PROCNAME, file $FILENAME." >> $LOGFILE
#
```

```
 $ORACLE_HOME/bin/sqlplus -s gfo@develop/t2gtgg @$BINDIR/$SCRIPTNAME > $SCRIPTOUTPUT
#
# check success of procedure
#
 SCRIPTERROR=$?
 cat $SCRIPTOUTPUT | grep 'SP2-|ORA-'
 GREPOUTPUT=$?
 if [ $GREPOUTPUT -eq 0 ]; then
   SCRIPTERROR=`expr $SCRIPTERROR + 1`
 fi
 if [ $SCRIPTERROR -ne 0 ]; then
   echo `date +"%D %T"`"\tproc_$PROJECT    \t" \
     "Error Running Procedure $PROCNAME, file $FILENAME." >> $LOGFILE
   exit 99
 fi
 rm $SCRIPTOUTPUT
 exit 0
else
 (handle_error) >> $LOGFILE
fi
#
# End of Processing
#
```

I.1.4 The GFO_SDR_OSC Table

```
#!/bin/sh
#
# load_gfo_sdr_osc_db.sh [FYTPE] [TEMPDIR] [FILENAME]
#=============================================================================
#
# Handle Errors
#
handle_error()
{
 mv $PROCDIR/$FILENAME $ERRORDIR/$FILENAME
 MAILF="/tmp/mail.$$"
 echo "ERROR - No config file exists in $CONFDIR for $FILENAME" > $MAILF
 echo "       Bad file has been moved into $ERRORDIR" >> $MAILF
 $BINMAIL -s "Bad file in $PROCDIR" "$OPERATOR" < $MAILF
 rm -f $MAILF
```

```
 echo `date +"%D %T"`"\tproc_config.sh \t" \
   "Error: Cannot process $FILENAME, " \
   " copied to $ERRORDIR."
 exit 1
}
#
#=============================================================================
#
# Main Routine
#
# Include globals
#
. /gen/gfo/dist/bin/config.incl
#
# Set arguments
#
FTYPE=$1
TEMPDIR=$2
FILENAME=$3
#
# Set project
#
ROOTDIR="/gen/gfo/dist"
PROJECT="wff"
PROCDIR="$ROOTDIR/$PROJECT/proc"
CONFDIR="$ROOTDIR/$PROJECT/config"
OUTDIR="$ROOTDIR/$PROJECT/out"
LOGDIR="$ROOTDIR/log"
LOADLOGFILE="$LOGDIR/$FILENAME.$$.log"
LOGFILE="$ROOTDIR/$PROJECT.log"
BADDIR="$ROOTDIR/$PROJECT/bad"
#
if [ "$FTYPE" = "gfoO" ]; then
 BASE=`basename $FILENAME`
 BASE1=`echo $FILENAME | awk -F. '{printf "%s",$1}'`
 PROJ=`echo $FILENAME | awk -F. '{printf "%s",substr($0,1,3)}'`
 EXT=`echo $FILENAME | awk -F. '{printf "%s",$2}'`
#
 CONTROL="gfo_sdr_osc.ctl"
#
```

```
 echo `date +"%D %T"`"\tproc_$PROJECT    \t" \
     "Loading Oracle with $CONTROL, file $FILENAME." >> $LOGFILE
#
# run SQL Loader to load the data into the database
#
 ( $ORACLE_HOME/bin/sqlldr gfo@develop/t2gtgg control="$BINDIR/$CONTROL"  \
  data="$FILENAME"  log="$LOADLOGFILE" 2>&1 ) > /dev/null
#
# Print results of database load by grepping the log file.
#
 egrep -e "File|Rows|Total" $LOADLOGFILE | pr | lp -dtopex2
#
# Check to see if bad file exists in the bin directory, if so, move to /gen/gfo/dist/wff/bad
# directory
#
 BADFILE="$BASE1.bad"
 if [ -f $BINDIR/$BADFILE ] ; then
   mv $BINDIR/$BADFILE $BADDIR/$BADFILE
 fi
 rm $LOADLOGFILE
#
# run the stored procedure,
#  load_gfo_sdr_osc_db, to update the data in the database
#
 SCRIPTNAME=run_load_gfo_sdr_osc_db.sh
 SCRIPTERROR=0
 PROCNAME=load_gfo_sdr_db
 SCRIPTOUTPUT=$LOGDIR/temp.$$.out
 cat /dev/null > $SCRIPTOUTPUT
#
# run procedure to move files from the temporary tables to the permanent tables
#
 echo `date +"%D %T"`"\tproc_$PROJECT    \t" \
    "Running Procedure $PROCNAME, file $FILENAME." >> $LOGFILE
#
 $ORACLE_HOME/bin/sqlplus -s gfo@develop/t2gtgg @$BINDIR/$SCRIPTNAME > $SCRIPTOUTPUT
#
# check success of procedure
#
 SCRIPTERROR=$?
```

```
 cat $SCRIPTOUTPUT | grep 'SP2-|ORA-'
 GREPOUTPUT=$?
 if [ $GREPOUTPUT -eq 0 ]; then
   SCRIPTERROR=`expr $SCRIPTERROR + 1`
 fi
 if [ $SCRIPTERROR -ne 0 ]; then
   echo `date +"%D %T"`"\tproc_$PROJECT    \t" \
      "Error Running Procedure $PROCNAME, file $FILENAME." >> $LOGFILE
   exit 99
 fi
 rm $SCRIPTOUTPUT
 exit 0
else
 (handle_error) >> $LOGFILE
fi
#
# End of Processing
#
```

I.2 Script for Extracting Data for Calibration Trend

```
#! /bin/sh
#
# run_gfo_cal_trend.sh [FTYPE] [TMPDIR] [FILENAME]
#
# Main Routine
#
# Include globals
#
. $HOME/bin/config.incl
#
# Set arguments
#
FTYPE=$1
TMPDIR=$2
FILENAME=$3
#
# Set project variables
#
PROJECT="wff"
PROCDIR="$ROOTDIR/$PROJECT/proc"
LOGDIR="$ROOTDIR/log"
LOGFILE="$LOGDIR/$PROJECT.log"
```

```
#
# read file for start/end dates
#
if [ -f $TMPDIR/$FILENAME ]; then
  read LINE < $TMPDIR/$FILENAME
  STARTDATE=`echo $LINE | awk -F: '{printf "%s",$1}'`
  STARTDATE=`expr $STARTDATE`
  ENDDATE=`echo $LINE | awk -F: '{printf "%s",$2}'`
  ENDDATE=`expr $ENDDATE`
  PNAME=`echo $LINE | awk -F: '{printf "%s",$3}'`
#
# set process parameters
#
  SCRIPTNAME=exec_run_cal_trend.sql
  SCRIPTOUTPUT=$LOGDIR/trend.$$.out
  cat /dev/null > $SCRIPTOUTPUT
#
# run procedure to create the cal trend report
#
  echo `date +"%D %T"`"\tproc_$PROJECT     \t" \
     "Running $SCRIPTNAME, for $STARTDATE, $ENDDATE, with $PNAME." >> $LOGFILE
#
  $ORACLE_HOME/bin/sqlplus -s gfo@develop/t2gtgg @$BINDIR/$SCRIPTNAME $STARTDATE $ENDDATE
\
"$PNAME" > $SCRIPTOUTPUT
#
# check success of procedure
#
  SCRIPTERROR=$?
  grep 'SP2- | ORA-' $SCRIPTOUTPUT
  GREPOUTPUT=$?
  if [ $GREPOUTPUT = 0 ]; then
     SCRIPTERROR=`expr $SCRIPTERROR + 1`
  fi
  if [ $SCRIPTERROR != 0 ]; then
     echo `date +"%D %T"`"\tproc_$PROJECT    \t" \
        "Error Running $SCRIPTNAME, file $PROCDIR/$FILENAME." >> $LOGFILE
  fi
  rm $SCRIPTOUTPUT
  mv $ROOTDIR/$PROJECT/out/*.cal $ROOTDIR/$PROJECT/in/.
  rm caltrendfile.sql
  exit 0
fi
```

```
exit 0
#
# End of processing
#
```

I.3 Script for Extracting Data for Engineering Trend

```
#! /bin/sh
#
# run_gfo_eng_trend.sh [FTYPE] [TMPDIR] [FILENAME]
#
# Main Routine
#
# Include globals
#
. $HOME/bin/config.incl
#
# Set arguments
#
FTYPE=$1
TMPDIR=$2
FILENAME=$3
#
# Set project variables
#
PROJECT="wff"
PROCDIR="$ROOTDIR/$PROJECT/proc"
LOGDIR="$ROOTDIR/log"
LOGFILE="$LOGDIR/$PROJECT.log"
#
# read file for start/end dates
#
if [ -f $TMPDIR/$FILENAME ]; then
  read LINE < $TMPDIR/$FILENAME
  STARTDATE=`echo $LINE | awk -F: '{printf "%s",$1}'`
  STARTDATE=`expr $STARTDATE`
  ENDDATE=`echo $LINE | awk -F: '{printf "%s",$2}'`
  ENDDATE=`expr $ENDDATE`
  PNAME=`echo $LINE | awk -F: '{printf "%s",$3}'`
#
# set process parameters
#
  SCRIPTNAME=exec_run_eng_trend.sql
  SCRIPTOUTPUT=$LOGDIR/trend.$$.out
```

```
  cat /dev/null > $SCRIPTOUTPUT
#
# run procedure to create the eng trend report
#
  echo `date +"%D %T"`"\tproc_$PROJECT     \t" \
     "Running $SCRIPTNAME, for $STARTDATE, $ENDDATE, with $PNAME." >> $LOGFILE
#
  $ORACLE_HOME/bin/sqlplus -s gfo@develop/t2gtgg @$BINDIR/$SCRIPTNAME $STARTDATE $ENDDATE
\
"$PNAME" > $SCRIPTOUTPUT
#
# check success of procedure
#
  SCRIPTERROR=$?
  grep 'SP2- | ORA-' $SCRIPTOUTPUT
  GREPOUTPUT=$?
  if [ $GREPOUTPUT = 0 ]; then
     SCRIPTERROR=`expr $SCRIPTERROR + 1`
  fi
  if [ $SCRIPTERROR != 0 ]; then
     echo `date +"%D %T"`"\tproc_$PROJECT     \t" \
        "Error Running $SCRIPTNAME, file $PROCDIR/$FILENAME." >> $LOGFILE
  fi
  rm $SCRIPTOUTPUT
  mv $ROOTDIR/$PROJECT/out/*.eng $ROOTDIR/$PROJECT/in/.
  rm engtrendfile.sql
  exit 0
fi
exit 0
#
# End of processing
#
```

I.4 Script for Extracting Data for Oscillator Drift Trend

```
#! /bin/sh
#
# run_gfo_eng_trend.sh [FTYPE] [TMPDIR] [FILENAME]
#
# Main Routine
#
# Include globals
#
. $HOME/bin/config.incl
```

```
#
# Set arguments
#
FTYPE=$1
TMPDIR=$2
FILENAME=$3
#
# Set project variables
#
PROJECT="wff"
PROCDIR="$ROOTDIR/$PROJECT/proc"
LOGDIR="$ROOTDIR/log"
LOGFILE="$LOGDIR/$PROJECT.log"
#
# read file for start/end dates
#
if [ -f $TMPDIR/$FILENAME ]; then
  read LINE < $TMPDIR/$FILENAME
  STARTDATE=`echo $LINE | awk -F: '{printf "%s",$1}'`
  STARTDATE=`expr $STARTDATE`
  ENDDATE=`echo $LINE | awk -F: '{printf "%s",$2}'`
  ENDDATE=`expr $ENDDATE`
  PNAME=`echo $LINE | awk -F: '{printf "%s",$3}'`
#
# set process parameters
#
  SCRIPTNAME=exec_run_eng_trend.sql
  SCRIPTOUTPUT=$LOGDIR/trend.$$.out
  cat /dev/null > $SCRIPTOUTPUT
#
# run procedure to create the eng trend report
#
  echo `date +"%D %T"`"\tproc_$PROJECT     \t" \
     "Running $SCRIPTNAME, for $STARTDATE, $ENDDATE, with $PNAME." >> $LOGFILE
#
  $ORACLE_HOME/bin/sqlplus -s gfo@develop/t2gtgg @$BINDIR/$SCRIPTNAME $STARTDATE $ENDDATE
\
"$PNAME" > $SCRIPTOUTPUT
#
# check success of procedure
#
  SCRIPTERROR=$?
  grep 'SP2-  | ORA-' $SCRIPTOUTPUT
```

```
  GREPOUTPUT=$?
  if [ $GREPOUTPUT = 0 ]; then
     SCRIPTERROR=`expr $SCRIPTERROR + 1`
  fi
  if [ $SCRIPTERROR != 0 ]; then
     echo `date +"%D %T"`"\tproc_$PROJECT     \t" \
        "Error Running $SCRIPTNAME, file $PROCDIR/$FILENAME." >> $LOGFILE
  fi
  rm $SCRIPTOUTPUT
  mv $ROOTDIR/$PROJECT/out/*.eng $ROOTDIR/$PROJECT/in/.
  rm engtrendfile.sql
  exit 0
fi
exit 0
#
# End of processing
#
```

I.5 Script for Extracting Data for GDR Trend

```
#! /bin/sh
#
# run_gfo_gdr_trend.sh [FTYPE] [TMPDIR] [FILENAME]
#
# Main Routine
#
# Include globals
#
. $HOME/bin/config.incl
#
# Set arguments
#
FTYPE=$1
TMPDIR=$2
FILENAME=$3
#
# Set project variables
#
PROJECT="wff"
PROCDIR="$ROOTDIR/$PROJECT/proc"
LOGDIR="$ROOTDIR/log"
LOGFILE="$LOGDIR/$PROJECT.log"
```

```
#
# read file for start/end dates
#
if [ -f $TMPDIR/$FILENAME ]; then
 read LINE < $TMPDIR/$FILENAME
 STARTCYCLE=`echo $LINE | awk -F: '{printf "%s",$1}'`
 STARTCYCLE=`expr $STARTCYCLE`
 ENDCYCLE=`echo $LINE | awk -F: '{printf "%s",$2}'`
 ENDCYCLE=`expr $ENDCYCLE`
 PNAME="gfoTrend"
#
# set process parameters
#
 SCRIPTNAME=exec_run_gdr_trend.sql
 SCRIPTOUTPUT=$LOGDIR/trend.$$.out
 cat /dev/null > $SCRIPTOUTPUT
#
# run procedure to create the ngdr trend report
#
 echo `date +"%D %T"`"\tproc_$PROJECT    \t" \
   "Running $SCRIPTNAME, for $STARTCYCLE, $ENDCYCLE, with $PNAME." >> $LOGFILE
#
 $ORACLE_HOME/bin/sqlplus -s gfo@develop/t2gtgg @$BINDIR/$SCRIPTNAME $STARTCYCLE $ENDCY-
CLE \
"$PNAME" > $SCRIPTOUTPUT
#
# check success of procedure
#
 SCRIPTERROR=$?
 grep 'SP2-' | ORA-' $SCRIPTOUTPUT
 GREPOUTPUT=$?
 if [ $GREPOUTPUT = 0 ]; then
   SCRIPTERROR=`expr $SCRIPTERROR + 1`
 fi
 if [ $SCRIPTERROR != 0 ]; then
   echo `date +"%D %T"`"\tproc_$PROJECT    \t" \
     "Error Running $SCRIPTNAME, file $PROCDIR/$FILENAME." >> $LOGFILE
 fi
 rm $SCRIPTOUTPUT
 mv $ROOTDIR/$PROJECT/out/$PNAME*.gdr $ROOTDIR/$PROJECT/in/.
 rm gdrtrendfile.sql
```

```
 exit 0
fi
exit 0
#
# End of processing
#
```

I.6 Script for Extracting Data for the GDR Cycle Report

```
#! /bin/sh
#
# run_gfo_gdr_cycle.sh [FTYPE] [TMPDIR] [FILENAME]
#
# Main Routine
#
# Include globals
#
. $HOME/bin/config.incl
#
# Set arguments
#
FTYPE=$1
TMPDIR=$2
FILENAME=$3
#
# Set project variables
#
PROJECT="wff"
PROCDIR="$ROOTDIR/$PROJECT/proc"
LOGDIR="$ROOTDIR/log"
LOGFILE="$LOGDIR/$PROJECT.log"
#
# read file for start/end dates
#
if [ -f $TMPDIR/$FILENAME ]; then
  read LINE < $TMPDIR/$FILENAME
  STARTCYCLE=`echo $LINE | awk -F: '{printf "%s",$1}'`
  STARTCYCLE=`expr $STARTCYCLE`
 #
 # set process parameters
 #
  SCRIPTNAME=exec_run_gdr_cycle.sql
  SCRIPTOUTPUT=$LOGDIR/cycle.$$.out
  cat /dev/null > $SCRIPTOUTPUT
 #
 # run procedure to create the ngdr cycle report
```

```
#
 echo `date +"%D %T"`"\tproc_$PROJECT    \t" \
   "Running $SCRIPTNAME, for cycle $STARTCYCLE." >> $LOGFILE
#
 $ORACLE_HOME/bin/sqlplus -s gfo@develop/t2gtgg @$BINDIR/$SCRIPTNAME $STARTCYCLE >
$SCRIPTOUTPUT
#
# check success of procedure
#
 SCRIPTERROR=$?
 grep 'SP2- | ORA-' $SCRIPTOUTPUT
 GREPOUTPUT=$?
 if [ $GREPOUTPUT = 0 ]; then
   SCRIPTERROR=`expr $SCRIPTERROR + 1`
 fi
 if [ $SCRIPTERROR != 0 ]; then
   echo `date +"%D %T"`"\tproc_$PROJECT    \t" \
     "Error Running $SCRIPTNAME, file $PROCDIR/$FILENAME." >> $LOGFILE
   exit 99
 fi
#
 SCRIPTNAME=exec_load_summary.sql
 PROCNAME=gfo_process_gdr_summary
#
# run procedure to update the summary table
#
 echo `date +"%D %T"`"\tproc_$PROJECT    \t" \
   "Running Procedure $PROCNAME, file $PROCDIR/$FILENAME." >> $LOGFILE
#
 $ORACLE_HOME/bin/sqlplus -s gfo@develop/t2gtgg @$BINDIR/$SCRIPTNAME $STARTCYCLE $START-
CYCLE >> $SC
RIPTOUTPUT
#
# check success of procedure
#
 SCRIPTERROR=$?
 grep 'SP2- | ORA-' $SCRIPTOUTPUT
 GREPOUTPUT=$?
 if [ $GREPOUTPUT = 0 ]; then
   SCRIPTERROR=`expr $SCRIPTERROR + 1`
 fi
 if [ $SCRIPTERROR != 0 ]; then
   echo `date +"%D %T"`"\tproc_$PROJECT    \t" \
     "Error Running Procedure $PROCNAME, file $PROCDIR/$FILENAME." >> $LOGFILE
 fi
#
 rm $SCRIPTOUTPUT
```

```
 mv $ROOTDIR/$PROJECT/out/gfoSycle*.gdr $ROOTDIR/$PROJECT/in/.
 rm gdrcyclefile.sql
 exit 0
fi
exit 0
#
# End of processing
#
```

Abbreviations & Acronyms

ADFC	Altimetry Data Fusion Center
CAL	Calibration Mode or Calibration Mode data
CAL/VAL	Calibration and Validation
CPU	Central Processing Unit
DSU	Digital Storage Unit
ENG	Engineering Data
ERO	Exact Repeat Orbit
FTP	File Transfer Protocol
GFO	GEOSAT Follow-On
GMT	Greenwich Mean Time
GSFC	Goddard Space Flight Center
HQ	Headquarters
HQ ISCS	Integrated Satellite Control System (NAVSOC's ground system at HQ for controlling satellites)
HQ RDCC	Remote Doppler Collection Computer at HQ
HW	Hardware
IDL	Interactive Data Language
MOE	Medium-accuracy Orbit Ephemerides
NAVO	NAVOCEANO
NAVOCEANO	Naval Oceanographic Office
NAVSOC	Naval Satellite Operations Center
NCEP	National Centers for Enviromental Prediction
NGDR	NOAA Geophysical Data Record
NSI	NASA Science Internet
OODD	Operational Orbit Determination Data
POC	Payload Operations Center
POE	Precision Orbit Ephemerides
RA	Radar Altimeter
SCC	Satellite Clock Coefficient

SCI	Science Data
SDR	Science Data Record
SDT	Science Definition Team
SW	Software
TRK	Track Mode
UTC	Coordinated Universal Time
VTCW	Vehicle Time Code Word
WF	Waveform Data
WFF	Wallops Flight Facility
WVR	Water Vapor Radiometer
XM	Transmitter

Other Documents in this Series

Volume 1	TOPEX Radar Altimeter Development Requirements and Specifications, Version 6.0, August 1988 (Published May 2003)
Volume 2	WFF Topex Software Documentation Overview, May 1999 (Published May 2003)
Volume 3	WFF TOPEX Software Documentation Altimeter Instrument File (AIF) Processing, October 1998 (Published July 2003)
Volume 4	TOPEX SDR Processing, October 1998 (Published July 2003)
Volume 5, Rev. 1	TOPEX GDR Processing, July 2003
Volume 5, Ver. 1.0	GFO Radar Altimeter Processing at Wallops Flight Facility, August 2003

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	August 2003	Technical Memorandum

4. TITLE AND SUBTITLE

GEOSAT Follow-On (GFO) Altimeter Document Series
GFO Radar Altimeter Processing at Wallops Flight Facility

5. FUNDING NUMBERS

Code 972

6. AUTHOR(S)

Dennis W. Lockwood, A.M. Conger

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS (ES)

Raytheon ITSS
GSFC Wallops Flight Faciity

Wallops Island, VA 23337

8. PEFORMING ORGANIZATION REPORT NUMBER

2003-03167-0

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS (ES)

National Aeronautics and Space Administration
Washington, DC 20546-0001

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

TM-2003-212236,
Vol. 5, Ver. 1.0

11. SUPPLEMENTARY NOTES

Dennnis W. Lockwood, A.M. Conger, Raytheon ITSS, Wallops Island, VA.

12a. DISTRIBUTION / AVAILABILITY STATEMENT

Unclassified–Unlimited
Subject Category:
Report available from the NASA Center for AeroSpace Information,
7121 Standard Drive, Hanover, MD 21076-1320. (301) 621-0390.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

This document is a compendium of the WFF GFO Software Development Team's knowledge regarding of GDO CAL/VAL Data. It includes many elements of a requirements document, a software specification document, a software design document, and a user's guide. In the more technical sections, this document assumes the reader is familiar with GFO and its CAL/VAL Data.

14. SUBJECT TERMS

GFO CAL/VAL

15. NUMBER OF PAGES
206

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL